

SKRIPSI

IMPLEMENTASI METODE *HIERARCHICAL AGGLOMERATIVE CLUSTERING* UNTUK *CLUSTERING* DOKUMEN SKRIPSI BERDASARKAN KESAMAAN LINGUISTIK (Studi Kasus : Prodi Informatika Unkhair)



OLEH
Widya Maulinda Hi Arsad
07351811042

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KHAIRUN
TERNATE
2024**

LEMBAR PENGESAHAN

IMPLEMENTASI METODE *HIERARCHICAL AGGLOMERATIVE CLUSTERING*
UNTUK *CLUSTERING* DOKUMEN SKRIPSI BERDASARKAN KESAMAAN
LINGUISTIK (Studi Kasus : Prodi Informatika Unkhair)

Oleh
Widya Maulinda Hi Arsad
07351811042

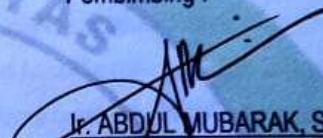
Skripsi ini telah disahkan
Tanggal 1 Maret 2024

Menyetujui
Tim Penguji

Ketua Penguji


Ir. AMAL KHAIIRAN, S.T., M.Eng., IPM.
NIP. 197401112003121003

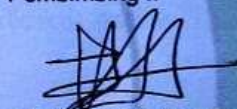
Pembimbing I


Ir. ABDUL MUBARAK, S.Kom., M.T., IPM.
NIP. 198212062014041002

Anggota Penguji


HAIRIL KURNIADI SIRAJUDDIN, S.Kom., M.kom.
NIP. 198204272023211009

Pembimbing II


MUHAMMAD FHADLI, S.Kom., M.Sc.
NIP. 199611232023211012

Anggota Penguji


Ir. SALKIN LUTE, S.Kom., M.T.
NIP. 198601112014041002

Mengetahui/Menyetujui

Koordinator Program Studi
Informatika


ROSIHAN S.T., M.Cs.
NIP. 197607192010121001


Dekan Fakultas Teknik
Universitas Khairun
Ir. ENDAH HARISUN, S.T., M.T., CRP.
NIP. 197511302005011013

LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini:

Nama	: Widya Maulinda Hi Arsad
NPM	: 07351811042
Fakultas	: Teknik
Jurusan/Program Studi	: Informatika
Judul	: Implementasi Metode <i>Hierarchical Agglomerative Clustering</i> Untuk <i>Clustering</i> Dokumen Skripsi Berdasarkan Kesamaan Linguistik (Studi Kasus Prodi Informatika Unkhair)

Dengan ini menyatakan bahwa penulisan Skripsi yang telah saya buat ini merupakan hasil karya sendiri dan benar keasliannya. Apabila ternyata di kemudian hari penulisan Skripsi ini merupakan hasil plagiat atau penjiplakan terhadap karya orang lain, maka saya bersedia mempertanggung jawabkan sekaligus bersedia menerima sanksi berdasarkan aturan tata tertib di Universitas Khairun.

Demikian pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.

Penulis



Widya Maulinda Hi Arsad

HALAMAN PERSEMBAHAN

Bismillahirrahmannirrahim

Dengan rahmat Allah SWT yang maha pengasih lagi maha penyayang dengan ini saya persembahkan skripsi ini untuk

Keluarga tercinta terumata mama, alm.papa, ata, kaka dan winda terima kasih atas limpahan doa dan kasih sayang yang tak terhingga dan selalu memberikan dorongan baik secara moril maupun material sehingga dapat menyelesaikan skripsi ini.

Dosen-dosen dan staf pengurus prodi informatika yang telah membagikan ilmu dan pengalamannya hingga penulis dapat sampai ke tahap ini.

Teman-teman angkatan 2018, terkhususnya Vivi, Ika, Amalia, Kiki, Isti dan Pragos yang selalu memberikan warna dalam perkuliahan, selalu memberikan dukungan dan semangat untuk sampai ke tahap ini.

MOTTO

"One day, I'm gonna have everything I prayed for. I really believe it"

"Trust That What Belongs To You Will Always Find You

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah Subhanahu wata'ala yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga penulis mampu menyelesaikan laporan skripsi "*Implementasi Metode Hierarchical Agglomerative Clustering Untuk Clustering Dokumen Skripsi Berdasarkan Kesamaan Linguistik (Studi Kasus : Prodi Informatika Unkhair)*".

Penyusunan laporan skripsi ini adalah untuk memenuhi salah satu persyaratan kelulusan pada Universitas Khairun Ternate Fakultas Teknik Program Studi Informatika. Penyusunan skripsi ini dapat terlaksana dengan baik berkat dukungan dari banyak pihak, untuk itu pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Dr., M. Ridha Ajam, M.Hum., selaku Rektor Universitas Khairun Ternate.
2. Bapak Ir., Endah Harisun, S.T., M.T., CRP., selaku Dekan Fakultas Teknik Universitas Khairun Ternate.
3. Bapak Rosihan, S.T., M.Cs., selaku Koordinator Program Studi Informatika.
4. Bapak Ir. Abdul Mubarak, S.Kom., M.T., IPM., selaku Pembimbing I yang telah memberikan arahan dan bimbingannya selama penulis melakukan penelitian dan penyelesaian skripsi ini.
5. Bapak Muhammad Fhadli, S.Kom., M.Cs., selaku Pembimbing II yang selalu memberikan motivasi, kritik dan saran yang membangun selama penyusunan skripsi ini.
6. Para Dosen Program Studi Informatika Universitas Khairun Ternate yang telah memberikan bekal ilmu kepada penulis.
7. Orang Tua Tercinta, terkhususnya Mama dan Alm.Papa atas segala bentuk doa, dukungan, dan motivasi yang diberikan kepada penulis. Terima kasih untuk kedua kakak Kak Agung dan Kak Ewi serta adik Winda atas dukungannya selama ini.
8. Teman-teman seperjuangan terkhususnya Vivi, Kiki, Amalia, Ika, Isti, Alwia, Andika dan Pragos atas dukungan dan bantuannya selama ini.
9. Seluruh angkatan 2018 yang telah memberikan bantuan dan dukungannya dalam penyelesaian skripsi ini.
10. Sahabat-sahabat tersayang penulis terkhususnya Widy, Ia, Hilya, Tiara, Eni, Eri, Eka dan Apit yang telah memberikan semangat dan motivasi dalam penyelesaian Skripsi

ini

11. Sobat-sobat tercinta Efa, Tia, Windy, Opi dan Nur yang selalu menguatkan dan memberi dukungan dalam segala situasi.
12. Semua pihak yang penulis tidak dapat sebutkan satu persatu yang telah membantu penulis baik langsung maupun tidak langsung dalam menyelesaikan laporan skripsi ini.

Penulis menyadari sepenuhnya bahwa laporan skripsi ini masih jauh dari sempurna, untuk itu semua jenis saran, kritik, dan masukan yang bersifat membangun dari semua pihak terkait sangat penulis harapkan. Akhir kata semoga penyusunan laporan skripsi ini dapat memberikan manfaat dan wawasan tambahan bagi para pembaca dan khususnya bagi penulis sendiri.

Ternate, 1 Maret 2024

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
HALAMAN PERNYATAAN KEASLIAN	iii
HALAMAN PERSEMBAHAN	iv
KATA PENGANTAR.....	v
DAFTAR ISI	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	xi
ABSTRAK.....	xii
 BAB I PENDAHULUAN	
1.1. Latar belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	2
1.4. Tujuan Penelitian.....	2
1.5. Manfaat Penelitian.....	2
1.6. Sistematika Penulisan	3
 BAB II TINJAUAN PUSTAKA	
2.1. Penelitian Terkait.....	4
2.2. <i>Clustering</i>	
2.2.1. <i>Hierarchical Agglomerative Clustering</i>	8
2.2.3. <i>Complete Linkage</i>	9
2.2.4 <i>Silhouette Coeficent</i>	10
2.3. <i>Data Mining</i>	11
2.3.2. CRISP-DM.....	12
2.3.3. <i>Term Frequency (tf) – Inverse Document Frequency (idf)</i>	13
2.3.4. <i>Preprocessing Text</i>	14
2.4. Skripsi	16
2.5. <i>Python</i>	16

2.6. <i>Library</i>	17
2.7. <i>Flowchart</i>	18

BAB III METODE PENELITIAN

3.1. Objek dan Waktu Penelitian	20
3.2. Alat dan Bahan Penelitian	20
3.2.1. Spesifikasi Perangkat Keras.....	20
3.2.2. Spesifikasi Perangkat Lunak	20
3.3. Metode Pengumpulan Data.....	21
3.4. Tahapan Penelitian	21
3.5. <i>Flowchart</i> Algoritma HAC	23
3.6. Contoh Perhitungan Metode <i>Hierarchical Agglomerative Clustering</i>	25
3.7. Contoh Penggunaan <i>Library</i>	27
3.8. Contoh Penggunaan <i>Silhouette Coeficient</i>	31

BAB IV HASIL DAN PEMBAHASAN

4.1. Analisis Data	32
4.2. Membaca Data dari File	33
4.3. Menampilkan beberapa baris data	33
4.4. <i>Preprocessing Text</i>	33
4.5. <i>Preprocessing Data</i>	34
4.6. Hasil <i>Clustering</i> Program	35
4.7. Grafik.....	40
4.7.1. <i>Scatter Plot</i> 2 Dimensi	40
4.7.2. <i>Scatter Plot</i> 3 Dimensi	42
4.8. Dendogram	43
4.9. <i>Word Cloud</i>	45
4.10. Implementasi Kesamaan Linguistik.....	46
4.11. Pengujian <i>Sillhouete Score</i>	48

BAB V KESIMPULAN

5.1. Kesimpulan	49
5.2. Saran.....	49

DAFTAR PUSTAKA

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Contoh Penggunaan <i>Python</i>	17
Gambar 3.1. <i>Flowchart</i> Tahapan Penelitian	21
Gambar 3.2. <i>Flowchart</i> Algoritma HAC.....	23
Gambar 3.3. Dendogram dari Perhitungan <i>Hierarchical Agglomerative Clustering</i>	27
Gambar 3.4. Contoh Penggunaan Library <i>Numpy</i>	27
Gambar 3.5. Contoh Penggunaan Library <i>Pandas</i>	28
Gambar 3.6. Contoh Penggunaan Library <i>Scikit Learn</i>	29
Gambar 3.7. Contoh Penggunaan Library <i>NLTK</i>	29
Gambar 3.8. Contoh Penggunaan Library <i>Sastrawi</i>	30
Gambar 3.9. Contoh Penggunaan Library <i>Matplotlib</i>	30
Gambar 3.10. Contoh Penggunaan <i>Silhouette Coeficient</i>	31
Gambar 4.2. Membaca Data dari file	33
Gambar 4.3. Menampilkan beberapa baris data.....	33
Gambar 4.4. <i>Preprocessing Text</i>	34
Gambar 4.5. Pengelompokkan Data	35
Gambar 4.6. Hasil Pencarian “Audit”	35
Gambar 4.7. Hasil Pencarian “Analisis”	36
Gambar 4.8. Hasil Pencarian “Deteksi” dan “Evaluasi”	36
Gambar 4.9. Hasil Pencarian “Sistem Pendukung Keputusan”	37
Gambar 4.10. Hasil Pencarian “Sistem Informasi Geografis”	37
Gambar 4.11. Hasil Pencarian “Sistem Informasi”	38
Gambar 4.12. Hasil Pencarian “Implementasi”	39
Gambar 4.13. Hasil Pencarian “Sistem Pakar”	39
Gambar 4.14. Hasil Pencarian “Sistem Pendukung Keputusan”	40
Gambar 4.15. <i>Scatter Plot</i> 2 Dimensi	41
Gambar 4.16. <i>Scatter Plot</i> 3 Dimensi	42
Gambar 4.17. Visualisasi Data Dalam Bentuk Dendogram	45
Gambar 4.18. Visualisasi Data Dalam Bentuk <i>World Cloud</i>	46

Gambar 4.19. Implementasi Kesamaan Linguistik	47
---	----

DAFTAR TABEL

	Halaman
Tabel 2.1. Perbandingan dengan Penelitian Terkait.....	4
Tabel 2.2. Simbol-simbol pada <i>flowchart</i>	18
Tabel 3.1. Spesifikasi Perangkat Keras (<i>Hardware</i>).....	20
Tabel 3.2. Spesifikasi Perangkat Lunak (<i>Software</i>).....	21
Tabel 3.3. Proses TF-IDF	25
Tabel 3.4. Hasil TF-IDF	25
Tabel 3.5. Penggabungan Dua <i>Cluster</i>	26
Tabel 4.1. <i>Dataset Judul Skripsi</i>	32

ABSTRAK

IMPLEMENTASI METODE *HIERARCHICAL AGGLOMERATIVE CLUSTERING* UNTUK *CLUSTERING* DOKUMEN SKRIPSI BERDASARKAN KESAMAAN LINGUISTIK (Studi Kasus : Prodi Informatika Unkhair)

Widya Maulinda Hi Arsad¹, Abdul Mubarak², Muhammad Fhadli³,
Program Studi Informatika, Fakultas Teknik, Universitas Khairun
Jl. Jati Metro, Kota Ternate

E-mail: widyamaulinda34@gmail.com¹, abdulmubarak@gmail.com², mfhadli@unkhair.ac.id³

Penelitian atau tugas akhir skripsi merupakan syarat kelulusan mahasiswa. Setiap tahun penelitian menjadi bertambah dan memungkinkan mahasiswa mengambil topik yang sama atau hampir serupa. Melalui penelitian yang dilakukan ini dikembangkan suatu aplikasi untuk mengcluster skripsi mahasiswa yang dapat berfungsi untuk membantu pihak jurusan dalam menyeleksi penelitian yang akan dilakukan mahasiswa, dan sebagai bahan rujukan bagi mahasiswa yang ingin mengambil judul penelitian. Proses *clustering* skripsi ini dilakukan dengan menggunakan metode *Hierarchical Agglomerative Clustering* pada sekumpulan judul skripsi dengan mengambil judul skripsi sebagai informasi yang dapat mewakili isi dokumen. Judul akan melewati proses preprocessing dengan menggunakan metode text mining. Setelah data judul melewati tahap preprocessing, maka judul dapat dikelompokkan dengan menggunakan metode *Hierarchical Agglomerative Clustering*. Hasil dari penelitian ini adalah menciptakan suatu aplikasi yang dapat mengelompokkan skripsi dengan otomatis. Teknik pengujian kualitas cluster pada penelitian ini menggunakan Silhouette Coefficient. Nilai Silhouette Score dari aplikasi yang dibuat dengan menggunakan metode *Hierarchical Agglomerative Clustering* dengan 2 cluster adalah 0.0733. Nilai 2 dipilih sebagai nilai yang paling ideal karena memiliki nilai *silhouette* yang paling besar.

Kata Kunci : *Hierarchical Agglomerative Clustering, Skripsi, Clustering*

BAB I

PENDAHULUAN

1.1. Latar belakang

Perkembangan teknologi informasi pada zaman modern ini telah sampai kepada era elektronik, ditandai dengan semakin digunakannya teknologi berupa komputer dan jaringan internet sebagai sarana utama penyampaian informasi. Dokumen yang beredar di dunia maya terus tumbuh dan mungkin menjadi kurang efektif dalam pencarian dan penyajian informasi (Efendi, 2012).

Seiring dengan kemajuan ilmu pengetahuan dan teknologi mayoritas orang lebih memanfaatkan teknologi sebagai media untuk belajar dan mencari informasi. Seperti halnya mahasiswa dalam mencari topik skripsi.

Menjelang semester akhir, program studi merekomendasikan beberapa topik yang harus dipilih oleh mahasiswa sebagai bidang konsentrasi dalam penyusunan skripsi. Topik ini akan membantu mahasiswa dalam membuat judul skripsi, yang didalamnya termuat pula metode yang dipakai dalam penyusunan skripsi.

Setelah merekomendasikan beberapa topik skripsi kepada mahasiswa, program studi akan menerima pengajuan judul skripsi dari mahasiswa, tugas program studi adalah memilih judul mana yang akan diterima sebagai judul skripsi dan judul mana yang ditolak. Untuk menentukan apakah judul tersebut diterima atau tidak, ada 2 (dua) faktor yang menjadi pertimbangan program studi, yaitu: Kesesuaian metode dengan topik yang dipilih dan terdapat kesamaan antara judul yang diajukan dengan judul skripsi sebelumnya.

Untuk memastikan ke dua faktor di atas program studi perlu melakukan evaluasi dan kontrol terhadap judul skripsi sebelumnya, evaluasi disini adalah melihat kembali judul

skripsi sebelumnya apakah judul yang telah diajukan oleh mahasiswa telah sesuai atau belum dengan topik dan metode yang telah dipilih. Adanya evaluasi mengenai kesesuaian judul dengan topik dan metode dapat membantu dalam memastikan apakah skripsi yang akan disusun mahasiswa sesuai dengan program studinya atau tidak.

1.2. Rumusan Masalah

Berdasarkan latar belakang diatas, dapat dirumuskan permasalahan yang akan diteliti adalah bagaimana penerapan *clustering* dokumen skripsi dengan menggunakan metode *Hierarchical Agglomerative Clustering* pada Prodi Informatika Unkhair.

1.3. Batasan Masalah

Batasan masalah pada penelitian ini antara lain sebagai berikut.

1. Penelitian dilakukan di Prodi Informatika Unkhair.
2. Data yang digunakan yaitu data dokumen skripsi Mahasiswa Prodi Informatika Universitas Khairun.
3. Metode yang digunakan adalah *Hierachical Agglomerative Clustering*.

1.4. Tujuan Penelitian

Adapun tujuan penelitian adalah untuk mengetahui hasil *clustering* dokumen skripsi dengan menerapkan metode *Hierarchical Agglomerative Clustering*.

1.5. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Manfaat Praktis Penelitian ini dapat menjadi salah satu alternatif untuk *clustering* dokumen skripsi sehingga dapat mempermudah mahasiswa dalam proses pengajuan judul skripsi.
2. Manfaat Akademis Hasil penelitian ini dapat berguna bagi penelitan sejenis sehingga

dapat digunakan sebagai referensi untuk melakukan pengembangan lebih lanjut sehingga lebih baik.

1.6. Sistematika Penulisan

Dalam Penulisan skripsi ini menggunakan sistematika penulisan sebagai berikut.

BAB I PENDAHULUAN

Terdiri dari latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Memaparkan teori-teori yang didapat dari sumber-sumber yang relevan untuk digunakan sebagai panduan dalam penelitian serta penyusunan skripsi.

BAB III METODOLOGI PENELITIAN

Bab ini membahas tentang metode penelitian yang telah dilakukan oleh penulis dengan permasalahan yang diangkat.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan implementasi perancangan detail dalam hal kerja sistem beserta analisis terhadap sistem.

BAB V PENUTUP

Pada bab ini berisi kesimpulan dan saran dari keseluruhan laporan.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Pada penelitian ini terdapat beberapa sumber referensi serta acuan dari hasil penelitian sebelumnya yang berkaitan dengan penelitian yang akan dilakukan saat ini. Adapun beberapa penelitian terkait yang menjadi dasar penulisan pada penelitian ini antara lain dapat dilihat pada tabel 2.1.

Tabel 2.1 Perbandingan dengan Penelitian terkait

No	Nama dan Tahun	Judul	Hasil
1.	Danang Aditya Wicaksana, Putra Pandu Adikara, Sigit Adinugroho (2018)	<i>Clustering</i> Dokumen Skripsi Dengan Menggunakan <i>Hierarchical Agglomerative Clustering</i>	Metode <i>hierarchical agglomerative clustering</i> lebih sering menghasilkan <i>singleton</i> (<i>cluster</i> yang terdiri dari 1 dokumen) sehingga mempengaruhi ketepatan suatu <i>cluster</i> dalam mengelompokkan dokumen. Dari 3 parameter pemilihan jarak (<i>linkage</i>) dapat disimpulkan <i>average linkage</i> lebih baik dalam mengelompokkan dokumen dibandingkan dengan <i>single linkage</i> dan <i>complete linkage</i> . Sedangkan pada penelitian saat ini, dapat disimpulkan bahwa <i>complete linkage</i> adalah parameter pemilihan jarak yang paling efektif dalam <i>clustering</i> dokumen dibandingkan dua parameter lainnya.
2.	Irmayansyah, Siti Khaosaroh (2019)	Penerapan metode <i>hierarchical agglomerative clustering</i> berbasis <i>single linkage</i> untuk pengelompokan judul skripsi	Dari hasil data pengujian sistem informasi diperoleh hasil bahwa kelayakan sistem informasi yang dikembangkan memperoleh presentase kelayakan sebesar 82% yang berarti masuk dalam kategori sangat layak, persentase tersebut di peroleh dengan cara membandingkan jumlah total skor yang diobservasi dengan jumlah total skor yang diharapkan. Sedangkan untuk penelitian ini menggunakan metode <i>clustering</i> berbasis <i>complete linkage</i> yang lebih efektif dibandingkan dengan metode <i>single linkage</i> . Metode <i>single linkage</i> memiliki kekurangan yaitu dimana interpretasi atau hierarki yang

			rumit, bergantung pada pemilihan teknik <i>intercluster similarity</i> yang lebih dikenal dengan istilah linkage serta komputasi yang mahal. Beberapa kelemahan dari <i>linkage</i> tersebut adalah sensitif terhadap adanya outlier, kesulitan menangani variasi bentuk dan ukuran, dan memisahkan <i>cluster</i> yang besar.
3.	Laili Cahyani, Muchamad Arif (2022)	<i>Text Mining</i> untuk Pengelompokan Skripsi di Prodi Pendidikan Informatika Universitas Trunojoyo Madura	Pada penelitian ini, telah diimplementasikan metode <i>K-means</i> untuk pengelompokan dokumen skripsi. Adapun nilai K Optimal yang didapatkan yaitu 3 cluster. Hasil penelitian ini menunjukkan bahwa metode clustering menggunakan <i>K-Means</i> dapat melakukan pengelompokan skripsi secara optimal dengan nilai akurasi sebesar 0,972972973, nilai presisi sebesar 916666667, nilai <i>recall</i> sebesar 0,9849199722, dan <i>F-Measure</i> sebesar 0,949199722 dalam skala 0 – 1. Tetapi, metode ini mempunyai kelemahan yaitu cukup sulit jika digunakan untuk mencari jarak dari data yang berdimensi banyak serta perlu inisialisasi nilai k menggunakan metode lain untuk mendapatkan nilai k yang optimal. Sedangkan pada penelitian ini menggunakan <i>Hierarchical Agglomerative Clustering</i> mengasumsikan setiap data yang ada sebagai <i>cluster</i> di awal proses. Jika jumlah data adalah n, dan jumlah <i>cluster</i> adalah k, maka besarnya $n = k$. Kemudian dihitung jarak antar clusternya dengan menggunakan <i>Euclidean distance</i> berdasarkan jarak rata-rata antar objek. Selanjutnya, dari hasil perhitungan jarak dipilih jarak yang paling minimal dan digabungkan sehingga besarnya $n = n - 1$. Ketika dua <i>cluster</i> digabungkan, jarak antara dua <i>cluster</i> yang digabungkan dengan <i>cluster</i> yang lain di-update. Penggabungan cluster akan terus dilakukan dan akan berhenti jika memenuhi kondisi jumlah $k = 1$. Pada akhir tahap <i>hierarchical clustering</i> diperoleh dendrogram yang menunjukkan urutan pengelompokan masing-masing anggota dalam <i>cluster</i> .
4.	Dezty Adhe Chajannah	Implementasi <i>Text Mining</i>	Hasil pengelompokan yang terbentuk dari dokumen skripsi menggunakan metode K-

	Rachman, Rito Goejantoro, dan Fidia Deny Tisna Amijaya (2020)	Pengelompokan Dokumen Skripsi Menggunakan Metode <i>K-Means Clustering</i>	<i>Means Clustering</i> adalah sebanyak 2 kelompok dengan anggota <i>cluster</i> ke-1 sebanyak 85 dokumen dan anggota <i>cluster</i> ke2 sebanyak 34 dokumen. Dokumen-dokumen skripsi yang masuk ke <i>cluster</i> 1 didominasi penelitian dengan metode data mining terutama tentang klasifikasi, analisis runtun waktu, analisis regresi, analisis data uji hidup, analisis spasial dan operasi riset. Sedangkan dokumen-dokumen skripsi yang masuk ke <i>cluster</i> 2 didominasi penelitian dengan metode analisis multivariat, pengendalian mutu dan matematika asuransi. Namun, <i>K-means</i> mempunyai kelemahan yang diakibatkan oleh penentuan pusat awal cluster. Hasil cluster yang terbentuk dari metode <i>K-means</i> ini sangatlah tergantung pada inisiasi nilai pusat awal <i>cluster</i> yang diberikan. Sedangkan dalam penelitian ini menggunakan metode <i>Hierarchical Agglomerative Clustering</i> yang nantinya hasil dari penelitian ini akan diketahui kemiripan atau kedekatan antar data sehingga dapat dikelompokkan ke dalam beberapa <i>cluster</i> , dimana antar anggota cluster memiliki tingkat kemiripan yang tinggi.
5.	Nengah Widya Utami, I Gede Juliana Eka Putra (2022)	<i>Text Mining Clustering</i> Untuk Pengelompokan Topik Dokumen Penelitian Menggunakan Algoritma <i>K-Means</i> Dengan Cosine Similarity	Berdasarkan hasil penelitian pengelompokan tema/topik dokumen Jurusan STMIK Primakara, jumlah kelompok optimal pada $k=6$. Hasil pengelompokan dapat dijadikan acuan untuk pelabelan kelompok topik. Topik penelitian yang dilakukan dosen di STMIK Primakara meliputi Pengembangan dan Evaluasi Sistem Informasi, <i>E-Government</i> , <i>Data Mining</i> , Teknologi Pendidikan, <i>Machine Learning/Artificial Intelligence</i> , serta Manajemen dan Bisnis. Pada penelitian diatas menggunakan data dan atribut yang kurang beragam karena hanya terdiri dari 52 dataset penelitian sedangkan pada penelitian ini menggunakan lebih dari 100 dataset penelitian sehingga mendapatkan hasil <i>cluster</i> yang lebih baik.
6.	Kitami Akromunnisa,	Klasifikasi Dokumen Tugas	Berdasarkan penelitian yang telah dilakukan didapatkan kesimpulan bahwa klasifikasi

	Rahmat Hidayat (2019)	Akhir (Skripsi) Menggunakan K-Nearest Neighbor	menggunakan metode <i>k-nearest neighbor</i> bisa digunakan untuk mengklasifikasi data intisari bahasa Indonesia dan judul dengan akurasi yang lebih besar tanpa melalui proses <i>stemming</i> . Untuk partisi data yang menggunakan pembagian data <i>Split into train test sets</i> dengan rasio perbandingan 9:1 menghasilkan akurasi lebih besar dibandingkan dengan rasio perbandingan 6:4, 7:3, 8:2 dan pembagian data menggunakan <i>kfold cross validation</i> . Dari hasil yang diperoleh maka dapat disimpulkan bahwa semakin besar data latih akan semakin baik akurasinya. Namun metode <i>k-nearest neighbor</i> memiliki kelemahan yaitu tidak berfungsi dengan baik pada dataset berukuran besar. Untuk dataset berukuran besar, <i>cost</i> untuk menghitung jarak antara titik baru dan setiap titik yang ada sangat besar dan cenderung menurunkan kinerja algoritma. Berbeda dengan penelitian diatas, penelitian saat ini menggunakan metode <i>hierarchical agglomerative clustering</i> yang mampu menggambarkan kedekatan antar data dengan dendrogram, cukup mudah untuk pembuatannya, serta dapat menentukan banyak cluster yang terbentuk setelah dendrogram terbentuk.
7.	Aditya Pradana dan Randy Ridwansyah (2021)	Klasifikasi Topik Skripsi Berdasarkan Makna dengan Pendekatan Semantik Web	Dari hasil penelitian yang dipaparkan di atas, ditemukan bahwa teknologi semantik web dapat membantu proses klasifikasi topik skripsi mahasiswa. Proses klasifikasi dilakukan secara aktual dan prediktif berdasarkan kata kunci dan makna yang terkandung di dalamnya. Hasil klasifikasi secara aktual menunjukkan bahwa skripsi mahasiswa TI Unpad didominasi oleh topik mengenai Sistem Informasi dan Multimedia, dengan presentase 50.23%. Sedangkan, klasifikasi prediktif dengan menggunakan algoritma KNN menghasilkan presentase 47.88%. Hasil yang berbeda tersebut diperoleh karena <i>Confusion Matrix</i> menunjukkan nilai sebagai berikut: AUC (0.711), CA (0.545), F1(0.578), <i>Precision</i> (0.669), <i>Recall</i> (0.545). Namun pada

			penelitian ini memiliki kelemahan karna menggunakan metode <i>k-nearest neighbor</i> yang sensitif terhadap data pencilan (<i>outlier</i>) dan tidak mengatasi nilai hilang (<i>missing value</i>) secara implisit. Berbeda dengan penelitian diatas, pada penelitian saat ini menggunakan metode <i>hierarchical agglomerative clustering</i> yang kelebihanannya mempercepat pengolahan data dan menghemat waktu karena data yang diinputkan akan membentuk hierarki atau membentuk tingkatan tersendiri sehingga mempermudah dalam penafsiran.
--	--	--	--

2.2. Clustering

Clustering adalah proses mengelompokkan atau penggolongan objek berdasarkan informasi yang diperoleh dari data yang menjelaskan hubungan antar objek dengan prinsip untuk memaksimalkan kesamaan antar anggota satu kelas dan meminimumkan kesamaan antar kelas/*cluster*. *Clustering* membagi data ke dalam grup-grup yang mempunyai objek yang karakteristiknya sama (Rahmawati, 2016).

2.2.1. Hierarchical Agglomerative Clustering

Hierarchical Agglomerative Clustering adalah *clustering* dengan pendekatan hirarki akan mengelompokkan data yang mirip dalam hirarki yang sama dan yang tidak mirip di hirarki yang agak jauh. Terdapat dua metode yang sering digunakan yaitu *agglomerative hierarchical clustering* dan *divisive hierarchical clustering*. *Agglomerative* melakukan *clustering* dari *N cluster* menjadi satu kesatuan *cluster*, dimana *N* adalah jumlah data, sebaliknya *divisive* melakukan proses *clustering* dari satu *cluster* menjadi *N cluster* (Everitt, 2011).

Dalam *Hierarchical Agglomerative Clustering* dokumen dikelompokkan secara berulang berdasarkan kemiripan yang terdekat antar data yang didasarkan dari

pembobotan kata (Jaiswal, 2011). Metode ini mengelompokkan suatu data dimulai dari data per-individu yang kemudian dikelompokkan hingga membentuk grup-grup dan proses pengelompokan secara berulang untuk menggabungkan grup-grup yang memiliki kemiripan hingga seluruh data terkelompokkan. Tahapan yang dilalui dalam proses HAC adalah sebagai berikut:

1. Menghitung bobot TF-IDF.
2. Menghitung nilai *Euclidean Distance* antar 2 dokumen.
3. Mencari nilai *Euclidean Distance* antar dokumen yang paling rendah.
4. Menggabungkan 2 dokumen dengan nilai *Euclidean Distance* terendah.
5. Memperbarui nilai *Euclidean Distance* dari gabungan data baru dengan data lain dengan parameter yang ditentukan (*single linkage/complete linkage/average linkage*).
6. Jika seluruh data belum tergabung menjadi 1 *cluster* maka kembali ke langkah 3.
7. Jika seluruh data sudah tergabung menjadi 1 *cluster* maka proses penggabungan selesai.

2.2.3. Complete Linkage

Complete linkage merupakan salah satu metode dari algoritma *hierarchical*. Algoritma *hierarchical* dimulai dengan setiap data yang dianggap sebagai *entity* yang berbeda. Untuk langkah pertama, dua data yang paling *similar* akan dikombinasikan kedalam sebuah *cluster*. Langkah selanjutnya adalah apakah dua data yang lain akan dikombinasikan untuk membentuk cluster kedua atau data ketiga akan ditambahkan ke *cluster* yang sudah ada dengan menggunakan jarak maksimal dari minimal (*max min operation*). Pada awal proses, setiap elemen berada dalam kelompoknya sendiri. Cluster tersebut kemudian digabungkan secara berurutan menjadi cluster yang lebih besar sampai semua elemen akhirnya beradada

di cluster yang sama.

2.2.4. *Euclidean Distance*

Euclidean Distance merupakan salah satu metode perhitungan jarak yang digunakan untuk mengukur jarak dari 2 (dua) buah titik dalam *euclidean space* (meliputi bidang *Euclidean* dua dimensi, tiga dimensi atau bahkan lebih). Rumus *Euclidean Distance* untuk mengukur tingkat kemiripan data dapat dilihat pada persamaan 2.1.

$$(x, y) = |x - y| = \sqrt{\sum_{i=1}^N (x_i - y_i)^2} \dots\dots\dots(2.1)$$

Keterangan:

$i=1$

d = jarak antara x dan y

x = data pusat kluster y = data pada atribut i = setiap data

n = jumlah data

x_i = data pada pusat kluster ke i

y_i = data pada setiap data ke i

2.2.4. *Silhouette Coeficent*

Metode ini merupakan metode evaluasi *cluster* yang menggabungkan metode *cohesion* dan *separation*. *Cohession* diukur dengan menghitung seluruh objek yang terdapat dalam sebuah *cluster* dan *separation* diukur dengan menghitung jarak rata-rata setiap objek dalam sebuah *cluster* dengan *cluster* terdekatnya (Rendon, 2011). Jarak antara data dihitung dengan menggunakan rumus *euclidean distance*. Untuk menyediakan informasi tentang kualitas hasil *clustering* pada proses *clustering*, dapat dihitung *silhouette* dari masing-masing *cluster* bahkan keseluruhan *cluster* dari hasil kerja suatu algoritma *clustering*. Rumus *Silhouette Coeficent* dapat dilihat pada persamaan 2.2.

$$sil(c) = sil(k) \frac{1}{|k|} \sum_{i=1}^1 sil(c_i) \dots\dots\dots(2.2)$$

Keterangan:

$sil(k)$: nilai *silhouette* semua *cluster*

$|k|$: banyaknya *cluster* k

$sil(c_i)$: rata-rata nilai *silhouette*

2.3. Data Mining

Data mining merupakan proses ataupun kegiatan untuk mengumpulkan data yang berukuran besar kemudian mengekstraksi data tersebut menjadi informasi-informasi yang nantinya dapat digunakan (Chairul, 2015). *Data mining* adalah suatu proses yang menggunakan teknik statistik, matematika, kecerdasan tiruan, dan *machine-learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai *database* besar. Istilah data mining memiliki hakikat sebagai disiplin ilmu yang tujuan utamanya adalah untuk menemukan, menggali, atau menambang pengetahuan dari data atau informasi yang kita miliki (Aziz, 2015).

Selain itu, *Data mining* merupakan analisis dari peninjauan kumpulan data untuk menemukan hubungan yang tidak diduga dan meringkas data dengan cara yang berbeda dengan sebelumnya, yang dapat dipahami dan bermanfaat bagi pemilik data. Ada beberapa teknik yang dimiliki data mining berdasarkan tugas yang bisa dilakukan, yaitu: deskripsi, estimasi, prediksi, klasifikasi, *clustering* dan asosiasi. Penelitian ini akan menggunakan fungsi klasifikasi dikarenakan sistem mengelompokan data didasarkan pada karakteristik alternatif. Dalam klasifikasi variabel, tujuan bersifat kategorik. Istilah *data mining* dan *knowledge discovery in databases* (KDD) sering kali digunakan secara bergantian untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar.

KDD dapat dibagi menjadi beberapa tahapan, antara lain: pembersihan data (*data cleaning*), integrasi data (*data integration*), seleksi data (*data selection*), transformasi data (*data transformation*), proses data mining, evaluasi pola (*pattern evaluation*).

2.3.1. Text Mining

Text mining adalah proses menemukan hal baru, yang sebelumnya tidak diketahui, mengenai informasi yang berpotensi untuk diambil manfaatnya dari sumber data yang tidak terstruktur mencakup dokumen bisnis, komentar *customer*, halaman web dan file XML. *Text mining* hampir sama dengan data mining dalam hal tujuan dan proses, tapi pada *text mining* inputnya adalah file data tidak terstruktur seperti dokumen dalam bentuk *word*, PDF, *text*, XML dan sebagainya. *Text mining* dapat digunakan dalam beberapa hal yaitu ekstraksi informasi, *topic tracking*, *summarization*, kategorisasi dan *clustering* (Kambey, 2020).

2.3.2. CRISP-DM

Cross-Industry Standard Process for Data Mining (CRISP-DM) adalah salah satu model atau *framework* dalam *data mining* yang awalnya (1996) dibangun oleh 5 perusahaan yaitu Integral Solutions Ltd (ISL), Teradata, Daimler AG, NCR Corporation dan OHRA. *Framework* ini kemudian dikembangkan oleh ratusan organisasi dan perusahaan di Eropa untuk dijadikan *methodology standard nonproprietary* bagi *data mining*. Menurut CRISP-DM, *data mining* memiliki siklus hidup yang terdiri dari enam fase, dan fase tersebut bersifat adaptif, yaitu fase berikutnya sangat bergantung pada hasil yang terkait dengan fase sebelumnya.

CRISP-DM (*Cross Industry Standard Process for Data Mining*) merupakan suatu standarisasi pemrosesan *data mining* yang telah dikembangkan dimana data yang ada akan melewati setiap fase terstruktur dan terdefinisi dengan jelas dan efisien. Selain menerapkan

suatu model dalam proses penambangan data, pemilihan algoritma sangat mempengaruhi terhadap komparasi kinerja metode *data mining*.

2.3.3. Term Frequency (tf) – Inverse Document Frequency (idf)

Metode ini merupakan metode untuk menghitung nilai/bobot suatu kata (*term*) pada dokumen. Metode ini akan mengabaikan setiap kata-kata yang tergolong tidak penting. Oleh sebab itu, sebelum melakukan metode ini, proses *stemming* dan *stopword removal* harus dilakukan terlebih dahulu oleh sistem. Karena melakukan pembobotan suatu kalimat bukan kata, pada metode ini terdapat 5 proses yang berbeda untuk perhitungan nilai suatu kalimat, yaitu (Budhi, 2008).

1. Kecocokan kata-kata pada kalimat dengan daftar kata kunci/*keyword*. Idenya adalah semakin tinggi nilai suatu kalimat, maka kalimat tersebut semakin penting keberadaannya di dalam suatu dokumen.
2. Menghitung frekuensi kata-kata suatu kalimat terhadap keseluruhan dokumen dan hasilnya akan dibagi dengan jumlah kata pada dokumen tersebut.
3. Bagian ketiga ini sangat sederhana yaitu hanya melihat posisi kalimat di dalam suatu paragraf. Berdasarkan metode deduktif induktif sesuai kaidah Bahasa Indonesia, ide pokok suatu paragraf terdapat pada kalimat yang berada di awal dan atau akhir dari paragraf tersebut.
4. Bagian keempat ini sangat berhubungan dengan hasil pemetaan dokumen. Pada bagian keempat ini akan dihitung jumlah relasi (yang disimbolkan dengan *edge*) suatu kalimat di dalam dokumen. Idenya adalah semakin banyak relasi yang dimiliki suatu kalimat dengan kalimat lainnya di dalam suatu dokumen maka kalimat tersebut kemungkinan mendiskusikan topik utama suatu dokumen.

5. Bobot kelima ini merepresentasikan seberapa penting sebuah kalimat dibandingkan dengan kalimat-kalimat lain yang terdapat pada semua dokumen yang akan diintegrasikan.

Setelah mendapatkan hasil dari kelima bobot diatas, selanjutnya nilai *tf* akan dihitung dengan persamaan 2.3.

$$tf = \text{bobot1} + \text{bobot2} + \text{bobot3} + \text{bobot4} + \text{bobot5} \dots\dots\dots (2.3)$$

Faktor lain yang diperhatikan dalam pemberian bobot adalah kejarang muncul kalimat (*sentence scarcity*) dalam koleksi. Kalimat yang muncul pada sedikit dokumen harus dipandang sebagai kata yang lebih penting (*uncommon sentences*) daripada kalimat yang muncul pada banyak dokumen. Pembobotan akan memperhitungkan faktor kebalikan frekuensi dokumen yang mengandung suatu kalimat (*inverse document frequency*).

Nilai dari *tf* akan dikalikan dengan nilai *idf* seperti pada persamaan 2.4 dan 2.5 (Intan, 2005):

$$w = tf \times idf \dots\dots\dots (2.4)$$

$$idf = \log \dots\dots\dots (2.5)$$

Keterangan :

W= bobot kalimat terhadap dokumen

tf = jumlah kemunculan kata/*term* dalam dokumen

N= jumlah semua dokumen yang ada dalam *database*

n= jumlah dokumen yang mengandung kata/*term*

idf = *inverse document frequency*

2.3.4. Preprocessing Text

Preprocessing Text adalah tahapan dimana pengguna melakukan seleksi data yang akan diolah menjadi lebih terstruktur. Pada tahapan *text preprocessing* tidak ada aturan atau

yang mengatur tahapan dari setiap proses yang harus dilakukan, semua tergantung pada jenis data yang akan diolah sehingga lebih terstruktur (Setiawan, 2020).

Adapun tahapan praproses teks diantaranya *Case Folding*, *Tokenizing*, *Stopwords*, dan *Stemming* (Ariadi, 2015).

1. *Case Folding*

Case folding adalah salah satu tahapan dalam *text preprocessing* yang paling sederhana dan efektif meskipun sering diabaikan. Tujuan dari *case folding* untuk mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya huruf “a” sampai “z” yang diterima (Firmansyah, 2020).

2. *Tokenizing*

Tahapan ini merupakan proses memecah yang semula berupa kalimat menjadi kata-kata. Proses pemecahan kalimat menjadi kata dilakukan berdasarkan spasi kata antar kalimat, sehingga menjadi kumpulan kata-kata dalam sebuah list yang disebut token.

3. *Stopwords*

Merupakan proses menghilangkan kosakata yang bukan merupakan kata unik atau kata yang tidak relevan, dan juga tidak bermakna seperti kata “dan”, “saya” yang ada pada *stop list library*.

4. *Stemming*

Stemming merupakan suatu proses dimana untuk mendapatkan *root/stem* atau kata dasar dari suatu kata dalam setiap kalimat dengan cara memisahkan masing-masing kata dari kata dasar dan imbuhan baik awalan (prefiks) maupun akhiran (sufiks). Sebagai contoh, kata bersama, kebersamaan, menyamai, yang dilakukan di stem ke root word nya yaitu “sama” (“Wahyudi, 2017). *Stemming* adalah teknik dalam NLP yang digunakan untuk

mengurangi kata-kata ke bentuk dasar atau akar kata.

2.4. Skripsi

Skripsi adalah suatu dokumen dari karya ilmiah yang disusun oleh mahasiswa pada tingkat strata 1 yang membahas suatu topik atau bidang tertentu dari hasil penelitian atau pengembangan yang telah dilakukan oleh mahasiswa tersebut guna mengikuti ujian akhir untuk memperoleh gelar sarjana (Huda, 2011). Setiap mahasiswa untuk memperoleh gelar sarjana dibutuhkan suatu dokumen skripsi. Sehingga skripsi merupakan suatu kewajiban yang harus dikerjakan untuk setiap mahasiswa strata 1.

2.5. Python

Bahasa pemrograman *python* di *release* pertama kali pada tahun 1991 oleh Guido van Rossum di Scitchting Mathematisch Centrum Belanda. *Python* dikembangkan bersifat *open source* dengan sebagian besar versinya menggunakan lisensi *GFL compatible*. Guido menggunakan nama *Python* karena ia adalah penggemar grup komedi Inggris bernama Monty Python (Wahyono, 2018).

Python memang muncul belakangan dibandingkan dengan Bahasa pemrograman *mainstream* semacam Bahasa C, *Visual Basic*, Java, atau PHP. Tetapi *Python* berkembang cukup pesat karena memiliki berbagai kelebihan seperti aspek *readability*, *multifungsi*, *interoperabilitas* dan juga dukungan komunitas yang memadai. Sampai dengan penelitian ini ditulis, versi *Python* yang telah di *release Python 3.9*.

2.5.1. Contoh Penggunaan Python

Dibawah ini adalah contoh penggunaan python dengan menggunakan metode *Hierarchical Agglomerative Clustering* dengan menggunakan dataset sederhana, dapat dilihat pada gambar 2.1.

```

from sklearn.datasets import make_blobs
from sklearn.cluster import AgglomerativeClustering
import matplotlib.pyplot as plt

# membuat data sintetis
data, _ = make_blobs(n_samples=50, centers=3, random_state=0)

# membuat objek model
model = AgglomerativeClustering(n_clusters=3)

# melakukan fitting pada data
model.fit(data)

# memprediksi label data
labels = model.labels_

# menampilkan hasil clustering
plt.scatter(data[:, 0], data[:, 1], c=labels)
plt.show()

```

Gambar 2.1 Contoh Penggunaan Python

2.6. Library

Terdapat beberapa *library* yang diperlukan untuk melakukan *clustering* dokumen dengan menggunakan metode *Hierarchical Agglomerative Clustering* antara lain sebagai berikut.

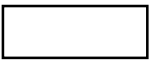
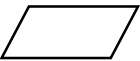
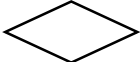
1. *Numerical Python (Numpy)* : berfungsi membantu menangani permasalahan angka-angka atau komputasi numerik di *Python*. *Numpy* merupakan *library Python* yang digunakan untuk pemrosesan *array*.
2. *Pandas* : Biasa digunakan untuk membuat tabel, mengecek data, mengubah dimensi, dan lain sebagainya. Struktur data dasar pada *Pandas* disebut *DataFrame*, hal ini memudahkan kita dalam membaca sebuah file dengan banyak jenis format seperti file *.txt*, *.csv*, dan *.tsv*.
3. *Scikit Learn* : *library* yang berfungsi untuk pemrosesan data maupun membangun model pembelajaran mesin (*training data*). *Library* ini memiliki berbagai algoritma pembelajaran, baik untuk regresi, pengelompokan, maupun klasifikasi. *Library* ini bekerja sama dengan *Numpy* dan *SciPy*.

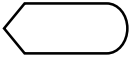
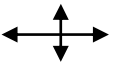
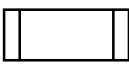
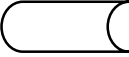
4. *Natural Language Toolkit (NLTK)* : *library* dan program pengolahan bahasa simbolik dan statistik alami (NLP) untuk bahasa inggris yang ditulis dalam bahasa Pemrograman *Python*. NLTK dimaksudkan untuk mendukung penelitian dan pengajaran di NLP yang termasuk ilmu *linguistic empiris*, ilmu kognitif, kecerdasan buatan, pencarian informasi, dan pembelajaran mesin.
5. *Sastrawi* : *library* yang dapat mengubah kata berimbuhan dalam bahasa Indonesia menjadi bentuk kata dasar. *Library* ini biasa digunakan pada *preprocessing text* bagian *stemming*.
6. *Matplotlib*: digunakan memplot angka-angka berdefinisi tinggi seperti diagram lingkaran, histogram, *scatterplot*, grafik, dan bentuk visualisasi lainnya.

2.7. Flowchart

Flowchart menggambarkan alur kerja dari suatu proses terhadap sistem yang telah dibuat agar mudah dipahami. *Flowchart* dijelaskan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses dengan proses lainnya dalam suatu program (Achlison, 2020). Simbol *Flowchart* bisa dilihat pada tabel 2.2.

Tabel 2.2 Simbol-simbol pada *flowchart*

No	Simbol	Fungsi
1		<i>Terminal, untuk memulai dan mengakhiri suatu proses / kegiatan.</i>
2		<i>Proses, suatu yang menunjukan setiap pengolahan yang dilakukan oleh komputer.</i>
3		<i>Input, untuk memasukan hasil dari suatu proses.</i>
4		<i>Decision, suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan</i>

5		<i>Display, output yang ditampilkan dilayar terminal</i>
6		<i>Connector, suatu prosedur akan masuk atau keluar melalui simbol ini dalam lembar yang sama.</i>
7		<i>Off Page Connector, merupakan simbol masuk atau keluarnya suatu prosedur pada kertas lembar lain.</i>
8		<i>Arus Flow, simbol ini digunakan untuk menggambarkan arus proses dari suatu kegiatan lain.</i>
9		<i>Hard Disk Storage, input output yang menggunakan hard disk.</i>
10		<i>Predifined Process, untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur.</i>
11		<i>Stored Data, input, output yang menggunakan disket.</i>
12		<i>Printer, simbol ini digunakan untuk menggambarkan suatu dokumen atau kegiatan untuk mencetak suatu informasi dengan mesin printer.</i>

BAB III

METODE PENELITIAN

3.1. Objek dan Waktu Penelitian

Pada penelitian ini objek yang digunakan dalam penelitian ialah data dari mahasiswa program Studi Informatika Universitas Khairun. Berdasarkan batasan masalah yang telah dibuat maka data yang diambil berupa data dokumen skripsi mahasiswa. Waktu penelitian dilakukan pada semester ganjil tahun ajaran 2022/2023.

3.2. Alat dan Bahan Penelitian

Dalam melakukan penelitian ini, terdapat beberapa spesifikasi alat penelitian yang peneliti gunakan. Spesifikasi alat penelitian ini merupakan standar minimal dari alat yang digunakan untuk mendukung proses analisa dalam penelitian ini. Alat yang digunakan dijelaskan sebagai berikut.

3.2.1. Spesifikasi Perangkat Keras

Spesifikasi kebutuhan perangkat keras dapat dilihat pada tabel 3.1.

Tabel 3.1 Spesifikasi Perangkat Keras (*Hardware*)

Jenis	Spesifikasi Hardware
Processor	Intel(R) Core(TM) i3-1115G4
<i>Installed Memory</i> (RAM)	4.00 GB
SSD	256 GB
OS	64-bit

3.2.2. Spesifikasi Perangkat Lunak

Spesifikasi kebutuhan perangkat lunak untuk penelitian ini dapat dilihat pada tabel 3.2 dibawah ini.

Tabel 3.2 Spesifikasi Perangkat Lunak (*Software*)

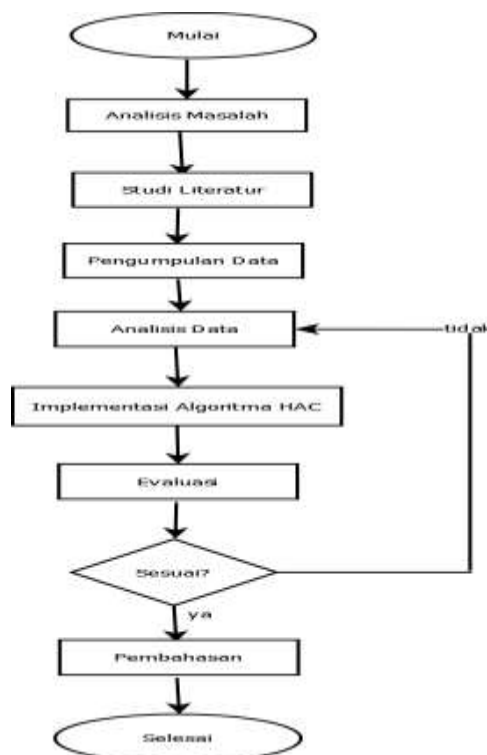
Jenis	Tipe	Keterangan
Sistem Operasi	Windows 11 Pro	Digunakan selama penelitian
Text Editor	<i>Visual Studio Code</i>	Untuk menulis dan mengedit skrip program yang dibuat
Bahasa Pemrograman	<i>Python</i> versi 3.10	Bahasa pemrograman yang digunakan untuk penulisan kode program

3.3. Metode Pengumpulan Data

Jenis data yang digunakan dalam penelitian ini adalah data primer. Data tersebut adalah data dokumen skripsi mahasiswa yang diambil pada prodi informatika Unkhair. Pada penelitian ini, penulis menggunakan 103 dataset.

3.4. Tahapan Penelitian

Tahapan penelitian merupakan langkah-langkah yang dilakukan oleh penulis dalam proses penelitian. Tahapan pada penelitian dapat dilihat pada gambar 3.1.

Gambar 3.1 *Flowchart* Tahapan Penelitian

1. Analisis Masalah

Tahapan pertama dalam penelitian ini adalah analisis masalah yang tujuannya untuk mengidentifikasi sejumlah masalah yang ada mengenai *clustering* dokumen skripsi.

2. Studi Literatur

Studi literatur yang dilakukan pada penelitian ini ialah mengumpulkan bahan referensi berupa jurnal, buku, dan berbagai referensi lainnya.

3. Pengumpulan Data

Pada tahapan ini peneliti melakukan proses pengumpulan data sebagai bagian yang paling utama dalam penelitian. Data yang dikumpulkan merupakan data skripsi mahasiswa yang diambil dari Prodi Informatika Universitas Khairun.

4. Analisis Data

Tahapan ini dimaksudkan untuk mengelompokan, melihat keterkaitan, membuat perbandingan, persamaan dan perbedaan atas data yang telah siap untuk dipelajari dan membuat model data dengan maksud untuk menemukan informasi yang bermanfaat sehingga dapat memberikan petunjuk untuk mengambil keputusan terhadap permasalahan penelitian yang dijalankan.

5. Implementasi Algoritma HAC

Pada tahapan ini dilakukan implementasi algoritma *Hierarchical Agglomerative Clustering* (HAC) yang diimplementasikan ke dalam program sederhana untuk melakukan *clustering* dokumen skripsi dengan menggunakan bahasa pemrograman *python*.

6. Evaluasi

Evaluasi dilakukan sebagai proses pengujian terhadap kinerja atau ketepatan proses *Clustering* Dokumen Skripsi dengan menggunakan metode *Hierarchical Agglomerative Clustering*. Tahapan ini dimaksudkan untuk mengevaluasi seberapa baik performa dari

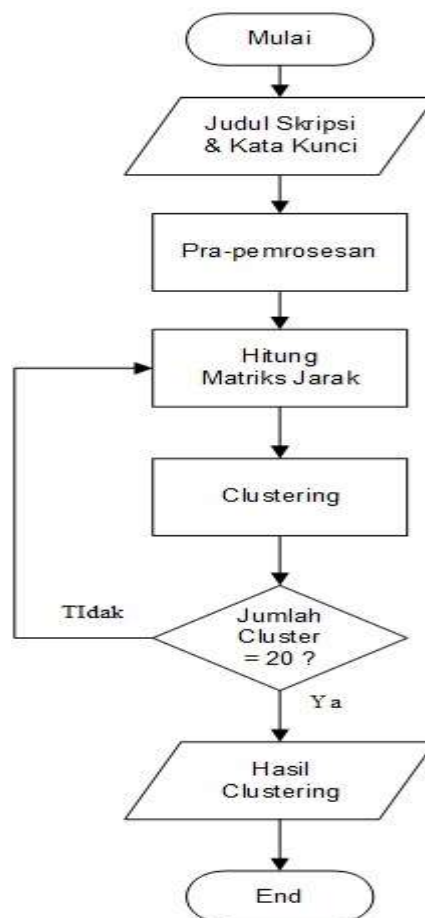
program yang dibuat melalui tahapan *preprocessing* text, TF-IDF, *Clustering* menggunakan metode HAC, dan yang terakhir *output* berupa hasil *clustering* dokumen oleh program.

7. Pembahasan

Tahapan ini merupakan tahapan terakhir yang bertujuan untuk mengulas hasil dari evaluasi metode *Hierarchical Agglomerative Clustering* dalam *clustering* dokumen skripsi.

3.5. Flowchart Algoritma HAC

Flowchart Algoritma HAC merupakan tahapan atau langkah-langkah program melakukan proses *clustering* dokumen dengan menggunakan metode atau algoritma *Hierarchical Agglomerative Clustering* (HAC). Lebih jelasnya dapat dilihat pada gambar 3.2.



Gambar 3.2 Flowchart Algoritma HAC

1. Masukan Judul Skripsi dan Kata Kunci

Tahapan Pertama dari kerja program yang akan dibangun diawali dengan menginput atau masukkan judul skripsi dan kata kunci. Pada tahapan ini judul skripsi dan kata kunci diperlukan untuk melakukan *preprocessing text*.

2. Pra-Pemrosesan

Pra-Pemrosesan atau *Preprocessing text* pada tahapan ini yaitu menyeleksi data *text* agar menjadi lebih terstruktur dengan melalui serangkaian tahapan yang meliputi tahapan *case folding*, *tokenizing*, *filtering* dan *stemming*.

3. Hitung matriks jarak

Pada tahapan ini dilakukan penggabungan dua konsep untuk perhitungan bobot yaitu, frekuensi kemunculan sebuah kata di dalam sebuah dokumen tertentu yang disebut *term frequency* (TF) dan *inverse* frekuensi dokumen yang mengandung kata yang disebut *inverse document frequency* (IDF).

4. Clustering HAC

Clustering dilakukan untuk proses mengelompokkan atau penggolongan objek berdasarkan informasi yang diperoleh dari data yang menjelaskan hubungan antar objek dengan prinsip untuk memaksimalkan kesamaan antar anggota satu kelas dan meminimumkan kesamaan antar kelas/*cluster* dengan objeknya yaitu dokumen skripsi mahasiswa dan proses *clustering* pada tahapan ini menggunakan algoritma *Hierarchical Agglomerative Clustering*. Pada penelitian menggunakan jumlah cluster sebanyak 20 *cluster*, jika jumlah *cluster* sudah sesuai maka dilanjutkan dengan melihat *clustering*.

5. Hasil Clustering

Tahapan ini merupakan tahapan terakhir berupa output dari program yang dibangun

yaitu hasil *clustering* judul skripsi menggunakan algoritma *Hierarchical Agglomerative Clustering*.

3.6. Contoh Perhitungan Metode *Hierarchical Agglomerative Clustering*

Dibawah ini adalah contoh perhitungan manual metode *Hierarchical Agglomerative Clustering* yang diawali dengan proses TF-IDF.

Kata Kunci : Pengelompokan Dokumen

Dokumen 1(D1) : Text Mining Untuk Pengelompokan Skripsi

Dokumen 2(D2) : Pengelompokan Seluruh Judul Skripsi Mahasiswa

Dokumen 3(D3) : Klasifikasi Dokumen Terjemahan

Jumlah Dokumen : 3

Proses TF-IDF dapat dilihat pada tabel 3.3.

Tabel 3.3. Tabel Proses TF-IDF

Token	kata kunci	tf			df	D/df	IDF(log D/df)	W			
		D1	D2	D3				kk	D1	D2	D3
Text	0	1	0	0	1	3	0.477	0	0.477	0	0
Mining	0	1	0	0	1	3	0.477	0	0.477	0	0
pengelompokan	1	1	1	0	2	1.5	0.176	0.176	0.176	0.176	0
Judul	0	0	1	0	1	3	0.477	0	0	0.477	0
klasifikasi	0	0	0	1	1	3	0.477	0	0	0	0.477
dokumen	1	0	0	1	1	3	0.477	0.477	0.477	0	0.477

Hasil TF-IDF dapat dilihat pada tabel 3.4.

Tabel 3.4. Hasil TF-IDF

objek	X1	X2	X3	X4	X5	X6
D1	0.477	0.477	0.176	0	0	0.477
D2	0	0	0.176	0.477	0	0
D3	0	0	0	0	0.477	0.477

Berdasarkan tabel 3.4 di atas maka kita lakukan perhitungan matematis dengan rumus *Euclidean distance* sebagai berikut:

1. Langkah Pertama, hitung matriks jarak dengan rumus *Euclidean*.

$$\begin{aligned}
 d_{D1D2} &= \sqrt{((0,477+0)^2 - (0,477-0)^2 - (0,176-0)^2 + (0-0,477)^2 + (0-477)^2 + (0-0,477)^2)} \\
 &= \sqrt{0,2275+0,2275+0+0,477+0+0,2275} \\
 &= \sqrt{1,595} \\
 &= 1,0768
 \end{aligned}$$

$$\begin{aligned}
 d_{D2D3} &= \sqrt{((0-0)^2 + (0-0)^2 + (0,176-0)^2 + (0-0,477)^2 + (0-477)^2 + (0-0,477)^2)} \\
 &= \sqrt{0+0+0,0309+0,2275+0,2275+0,2275} \\
 &= \sqrt{7134} \\
 &= 0,8446
 \end{aligned}$$

$$\begin{aligned}
 d_{D1D3} &= \sqrt{((0,477-0)^2 + (0,477-0)^2 + (0,176-0)^2 + (0-0)^2 + (0-0,477)^2 + (0,477-0,477)^2)} \\
 &= \sqrt{0,2275+0,2275+0,0309+0+0,2275+0} \\
 &= \sqrt{7134} \\
 &= 0,8446
 \end{aligned}$$

2. Langkah kedua, menggabungkan dua *cluster* terdekat yaitu *cluster* dengan D2 dengan D3 karena nilai jaraknya adalah 0.8446 yang paling kecil dibandingkan yang lainnya. Penggabungan cluster dapat dilihat pada tabel 3.5.

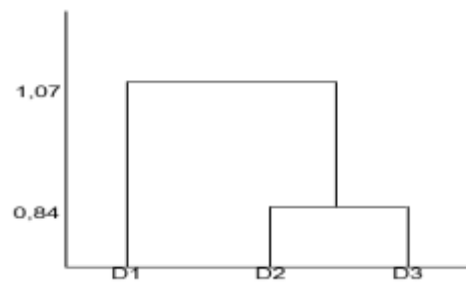
Tabel 3.5. Penggabungan Dua *Cluster*

	D23	D1
D23	0	1,0768
D1	1,0768	0

3. Langkah ketiga, kita akan memperbarui matriks jarak menggunakan teknik pengelompokan *complete linkage*.

$$D(23)D1 = \max\{d_{2;3}, d_{3;1}\} = \max\{1,0768; 0,8446\} = 1,0768$$

4. Langkah terakhir adalah membuat dendrogram sesuai anggota *cluster* yang terbentuk dan nilai jarak terdekatnya.



Gambar 3.3 Dendrogram dari Perhitungan *Hierarchical Agglomerative Clustering*

3.7. Contoh Penggunaan *Library*

Dibawah ini adalah contoh penggunaan beberapa *library* pada *python* yang digunakan dalam program yang akan dibuat.

1. *Numerical Python (Numpy)*

Contoh Penggunaan *Library Numpy* dapat dilihat pada Gambar 3.4.

```
#import library
import numpy as np
import matplotlib.pyplot as plt

#membuat variabel a berisi array dengan total 21 data bernilai 0 hingga 20
a = arange(21)

#membuat variabel b yang berisi fungsi sinus dari array a
b = sin(a)

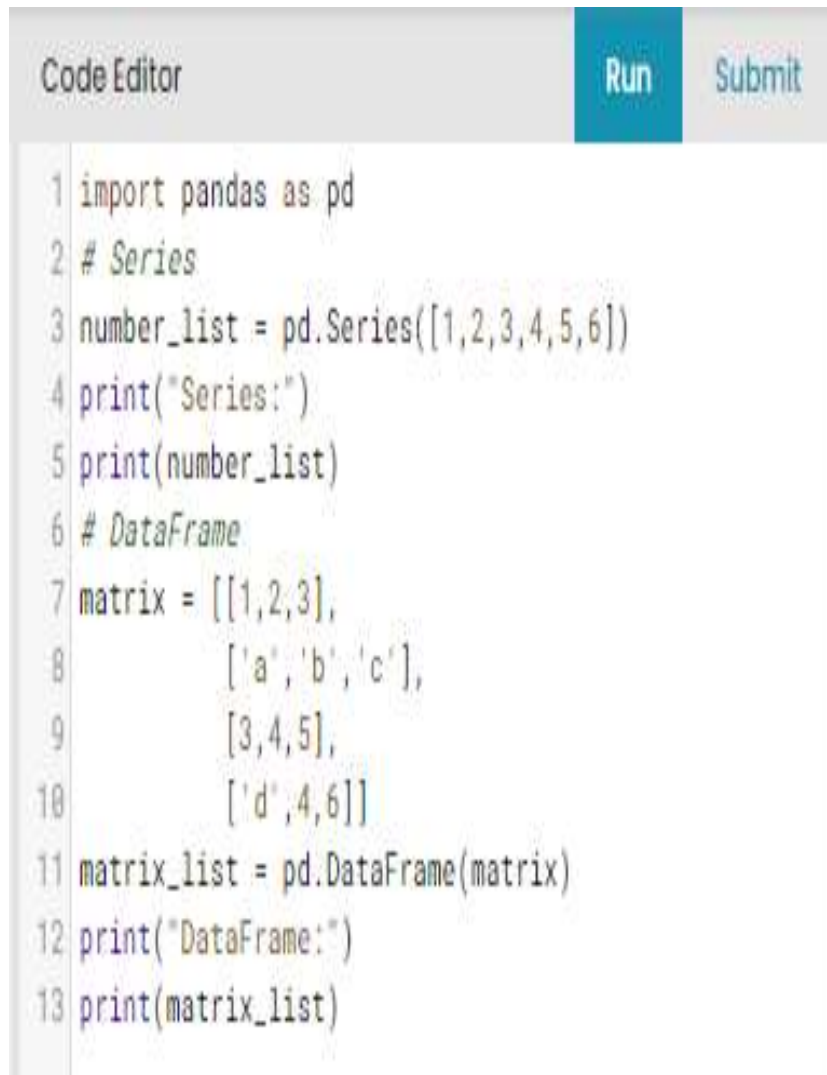
#menampilkan output berupa grafik
plt.plot(a,b)
plt.show()
```

Gambar 3.4 Contoh Penggunaan *Library Numpy*

Pada gambar 3.4 digunakan 2 *library* yaitu *numpy* untuk mengerjakan sistem matematis dan *Matplotlib* untuk membuat visualisasi. Dalam analisis data pada penelitian ini, *numpy* berperan sebagai penampung data yang diolah menggunakan algoritma dan *library*. Untuk data numerik, *array NumPy* lebih efisien untuk menampung dan mengolah data daripada bentuk data struktur lain di *python*.

2. *Pandas*

Contoh penggunaan library Pandas dapat dilihat pada gambar 3.5.



```

1 import pandas as pd
2 # Series
3 number_list = pd.Series([1,2,3,4,5,6])
4 print("Series:")
5 print(number_list)
6 # DataFrame
7 matrix = [[1,2,3],
8           ['a','b','c'],
9           [3,4,5],
10          ['d',4,6]]
11 matrix_list = pd.DataFrame(matrix)
12 print("DataFrame:")
13 print(matrix_list)

```

Gambar 3.5 Contoh Penggunaan *Library Pandas*

Gambar 3.5 diatas adalah contoh pembuatan *dataframe* sederhana yang menunjukkan nama dan usia dari sebuah kelompok menggunakan *library pandas*. Fungsi *import* yang bertujuan untuk mengaktifkan *library* yang ingin digunakan, dalam hal ini *Pandas* didefinisikan sebagai 'pd'. Sedangkan untuk penelitian ini, *pandas* berfungsi sebagai *library python* yang menyediakan struktur data dan fungsi tingkat tinggi yang di rancang untuk data terstruktur atau tabular dengan cepat, mudah dan ekspresif.

3. Scikit Learn

Contoh Penggunaan *Library Scikit Learn* dapat dilihat pada Gambar 3.6

```
from sklearn.datasets import make_blobs
from sklearn.cluster import AgglomerativeClustering
import matplotlib.pyplot as plt

# membuat data sintetis
data, _ = make_blobs(n_samples=50, centers=3, random_state=0)

# membuat objek model
model = AgglomerativeClustering(n_clusters=3)

# melakukan fitting pada data
model.fit(data)

# memprediksi label data
labels = model.labels_

# menampilkan hasil clustering
plt.scatter(data[:, 0], data[:, 1], c=labels)
plt.show()
```

Gambar 3.6 Contoh Penggunaan *Library Scikit Learn*

Diatas adalah contoh penggunaan *library scikit learn* menggunakan metode *Hierarchical Agglomerative Clustering* untuk *dataset* sederhana. Untuk penelitian ini, kelebihan *scikit learn* adalah kecepatannya saat melakukan tolok ukur yang berbeda dalam *dataset*.

4. NLTK

Contoh Penggunaan *Library NLTK* dapat dilihat pada Gambar 3.7.

```
# import library nltk
import nltk

# download corpus punkt
nltk.download('punkt')

from nltk.tokenize import word_tokenize
text = "Alhamdulillah, hari ini cuacanya cerah. Tapi, sore hari hujan"
print(word_tokenize(text))
```

Gambar 3.7 Contoh Penggunaan *Library NLTK*

Pada contoh diatas adalah penggunaan *library NLTK* untuk melakukan salah satu tahapan *preprocessing text* yaitu tokenisasi. Untuk penelitian ini, NLTK menyediakan komponen yang mudah digunakan secara mandiri tanpa dependensi dari sebuah *toolkit*.

5. Sastrawi

Contoh Penggunaan *Library Sastrawi* dapat dilihat pada Gambar 3.8.

```
# import StemmerFactory class
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

# create stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# stemming process
sentence = 'Perekonomian Indonesia sedang dalam pertumbuhan yang membanggakan'
output = stemmer.stem(sentence)

print(output)
# ekonomi indonesia sedang dalam tumbuh yang bangga

print(stemmer.stem('Mereka meniru-nirukannya'))
# mereka tiru
```

Gambar 3.8 Contoh Penggunaan *Library Sastrawi*

Diatas adalah contoh penggunaan *library sastrawi* untuk melakukan *stemming* atau proses mengubah kata berimbuhan menjadi kata dasar. Kelebihan *sastrawi* di penelitian ini yaitu kemampuan melakukan proses *stemming* dalam bahasa Indonesia.

6. Matplotlib

Contoh Penggunaan *Library Matplotlib* dapat dilihat pada gambar Gambar 3.9.

```
import matplotlib.pyplot as plt

#Input data sumbu X
x = [1, 2, 3, 4] #[x1,x2,x3,x4]

#Input data sumbu Y
y = [3, 5, 7, 9] #[y1,y2,y3,y4]

#Masukkan data ke line plot
plt.plot(x, y)

#Menentukan batas koordinat/axis
plt.axis((0,10,0,10)) #(x-min,x-max,y-min,y-max)

#Menampilkan plot
plt.show()
```

Gambar 3.9 Contoh Penggunaan *Library Matplotlib*

Pada contoh diatas adalah penggunaan *library matplotlib* untuk membuat sebuah *line plot* sederhana. Dalam penelitian ini *matplotlib* berperan untuk memvisualisasikan data agar lebih rapid dan tertata.

3.8. Contoh Penggunaan *Silhouette Coeficent*

Silhouette Coeficent digunakan untuk melihat kualitas dan kekuatan *cluster*, seberapa baik atau buruknya suatu obyek ditempatkan dalam suatu *cluster*. Dibawah ini adalah contoh penggunaan metode *Silhouette Coeficent* dengan mengimport beberapa *library python* ditambah dengan import *silhouette_score* untuk menguji kualitas dari *cluster*. Contoh Penggunaan *Sillhotte Coeficient* pada gambar 3.10.

```
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
%matplotlib inline
```

Gambar 3.10 Contoh Penggunaan *Silhouette Coeficent*

BAB IV

HASIL DAN PEMBAHASAN

4.1. Analisis Data

Pada bab ini membahas tentang bagaimana *clustering* judul skripsi dengan menggunakan metode *Hierarchical Agglomerative Clustering* (HAC) serta menentukan nilai *sillhouette coefficient* untuk menguji seberapa baik kekuatan *cluster* atau evaluasi *cluster*. Dataset yang digunakan adalah data dokumen skripsi mahasiswa prodi Informatika Universitas Khairun dari tahun 2021-2022 sebanyak 103 data yang bersumber langsung dari Tata Usaha Program Studi Informatika Universitas Khairun. Data yang digunakan terdiri dari teks judul skripsi dan atribut pendukung seperti kata kunci.

Berikut 3 contoh dataset dokumen skripsi yang digunakan pada penelitian ini dapat dilihat pada tabel 4.1.

Tabel 4.1. *Dataset* Judul Skripsi

	Judul Skripsi	Kata Kunci
1.	SISTEM INFORMASI AKADEMIK MADRASAH ALIYAH ALKHAIRAAT KOTA TERNATE BERBASIS WEB	Sistem Informasi, PHP, MySQL, Metode Prototype, Black Box
2.	SISTEM INFORMASI GEOGRAFIS PEMETAAN LOKASI KAFE DI KOTA TERNATE BERBASIS WEB	Pemetaan Lokasi Kafe, Ternate, Kafe, Sistem Informasi Geografis
3.	PENERAPAN METODE ANNALYTICAL HIERARCHY PROCESS (AHP) DALAM SISTEM PENDUKUNG KEPUTUSAN UNTUK PEMILIHAN EKSTRAKURIKULER PADA SISWA SMK 1 KOTA TERNATE	SPK , Annalytical Hierarchy Process, Pemilihan Ekstrakurikuler

Pada tabel 4.1. diatas terdapat 3 baris data dengan 2 kolom yaitu judul skripsi dan kata kunci. Namun data yang akan digunakan pada penelitian ini sebanyak 88 data yang telah ditentukan pada bab sebelumnya.

4.2. Membaca Data dari File

Pada tahap ini membaca data dari excel dengan nama file “dataset_judul_skripsi” untuk memulai proses *clustering*. Lebih jelasnya dapat dilihat pada gambar 4.1.

```
data = pd.read_excel('dataset_judul_skripsi.xlsx')
```

Gambar 4.1. Membaca data dari file

4.3. Menampilkan beberapa baris data

Tahap ini menampilkan beberapa baris data untuk memastikan pembacaan dataset berhasil, yang dapat dilihat pada gambar 4.2.

```
print(data.head())
```

```

                                judul_skripsi \
0  SISTEM INFORMASI AKADEMIK MADRASAH ALIYAH ALKH...
1  SISTEM INFORMASI GEOGRAFIS PEMETAAN LOKASI KAF...
2  PENERAPAN METODE ANNALYTICAL HIERARCHY PROCESS...
3  SISTEM PENDUKUNG KEPUTUSAN KELAYAKAN MENDIDIRI...
4  SISTEM PENDUKUNG KEPUTUSAN PERPANJANGAN MASA K...

                                kata_kunci
0  Sistem Informasi, PHP, MySQL, Metode Prototype...
1  Pemetaan Lokasi Kafe, Ternate, Kafe, Sistem I...
2  SPK , Annalytical Hierarchy Process, Pemilihan...
3  Kelayakan Izin Mendirikan Bangunan, Multy Obje...
4  PK, Multi Attribute Utility Theory, Pegawai te...
```

Gambar 4.2. Menampilkan beberapa baris data

4.4. *Preprocessing Text*

Pada tahap ini *Preprocessing Text* dilakukan dengan membangun matriks TF-IDF dari judul skripsi dan kata kunci, yang dapat dilihat pada gambar 4.3.

```
tfidf_vectorizer = TfidfVectorizer(stop_words='english')
tfidf_matrix = tfidf_vectorizer.fit_transform(data['judul_skripsi'], data['kata_kunci'])
```

Gambar 4.3. *Preprocessing Text*

4.5. *Preprocessing Data*

4.5.1. *Normalisasi Data*

Sebelum menerapkan metode HAC pada data judul skripsi, dilakukan normalisasi data untuk memastikan konsistensi dan keakuratan analisis. Normalisasi data dilakukan dengan langkah-langkah sebagai berikut:

1. Penghilangan Karakter Khusus

Karakter khusus, angka, dan simbol yang tidak relevan dihapus dari judul skripsi untuk memfasilitasi analisis teks.

2. Pengonversian Huruf

Semua huruf dalam judul skripsi dikonversi menjadi huruf kecil (lowercase) untuk menghindari perbedaan hasil yang disebabkan oleh perbedaan kapitalisasi.

4.5.2. *Seleksi Fitur*

Langkah ini melibatkan seleksi fitur untuk mengidentifikasi atribut yang paling relevan dalam *clustering* judul skripsi. Seleksi fitur dilakukan dengan menggunakan metode *term frequency-inverse document frequency* (TF-IDF) untuk menentukan bobot kata-kata dalam setiap judul skripsi.

4.5.3. *Pengelompokan Data*

Setelah normalisasi dan seleksi fitur, data disiapkan untuk proses pengelompokan menggunakan metode HAC. Matriks jarak antar-judul skripsi dihitung berdasarkan kesamaan bobot TF-IDF. Kemudian, metode HAC diterapkan untuk membentuk hierarki

cluster. Pengelempokkan data dapat dilihat pada gambar 4.4.

```
num_clusters = 8
hac_model = AgglomerativeClustering(n_clusters=num_clusters, metric='euclidean', linkage='ward')
# hac_model = AgglomerativeClustering(n_clusters=num_clusters)
clusters = hac_model.fit_predict(tfidf_matrix.toarray())
cluster_sizes = np.bincount(clusters)
```

Gambar 4.4. Pengelompokkan data

Pada gambar diatas, dilakukan penginputan jumlah *cluster* yaitu sebanyak 7 *cluster* serta penerapan algoritma *Hierarchical Agglomerative Clustering* ke dalam *source code*.

4.6. Hasil *Clustering* Program

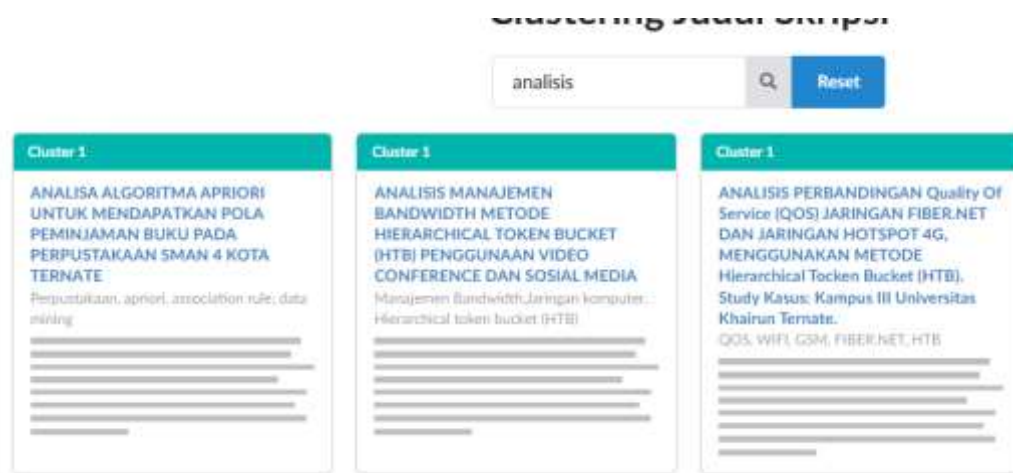
Pada tahapan ini terdapat hasil *clustering* judul skripsi menggunakan metode *Hierarchical Agglomerative Clustering*. Terdapat 6 topik judul skripsi yang paling banyak digunakan mahasiswa.



Gambar 4.5. Hasil Pencarian “Audit”

Pada gambar 4.5. topik “Audit” pada clu digunakan 2 mahasiswa sebagai judul skripsi dengan metode yang sama yaitu COBIT 5 dan studi kasus yang berbeda. Topik “Audit” juga dikelompokkan ke dalam cluster yang sama yaitu cluster 0. Cluster 0 ditandai dengan garis

warna coklat.



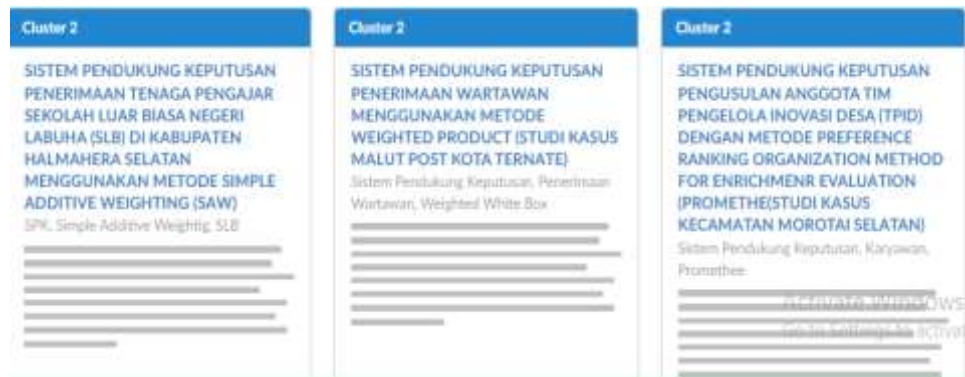
Gambar 4.6. Hasil Pencarian “Analisis”

Pada gambar 4.6.. topik “Analisis” digunakan 3 mahasiswa sebagai judul skripsi pada cluster 1. Pada gambar diatas topik “Analisis” digunakan 3 mahasiswa sebagai judul skripsi dengan masing-masing metode yang berbeda. Objek yang digunakan pada 3 judul diatas juga berbeda-beda antara satu dengan yang lainnya. Cluster 1 ditandai dengan garis warna hijau tosca.



Gambar 4.7. Hasil Pencarian “Deteksi” dan “Evaluasi”

Pada gambar 4.7. topik “Deteksi” dan juga “Evaluasi” masing-masing digunakan 1 mahasiswa sebagai judul skripsi. Kedua topik tersebut juga tergabung ke dalam cluster 1 yang banyak diminati mahasiswa. Cluster 1 ditandai dengan garis warna hijau tosca.



Gambar 4.8. Hasil Pencarian “Sistem Pendukung Keputusan”

Pada gambar 4.8. topik “Sistem Pendukung Keputusan” digunakan 3 mahasiswa sebagai judul skripsi yang terletak pada cluster 2. Pada gambar diatas topik “Sistem Pendukung Keputusan” digunakan 3 mahasiswa sebagai judul skripsi dengan masing-masing metode yang berbeda. Objek yang digunakan pada 3 judul diatas juga berbeda-beda antara satu dengan yang lainnya. Sistem Pendukung Keputusan adalah salah satu topik yang diminati banyak mahasiswa prodi Unkhair. Cluster 2 ditandai dengan garis warna biru.



Gambar 4.9. Hasil Pencarian “Sistem Informasi Geografis”

Gambar 4.9. Hasil Pencarian “Sistem Informasi Geografis” Pada gambar 4.7.4. topik “Sistem Informasi Geografis” digunakan 4 mahasiswa sebagai judul skripsi pada cluster 3. Pada gambar diatas topik “Sistem Informasi Geografis” digunakan 4 mahasiswa sebagai

judul skripsi dengan masing-masing metode yang berbeda. Objek yang digunakan pada 4 judul diatas juga berbeda-beda antara satu dengan yang lainnya. Cluster 3 ditandai dengan garis warna hijau toska.



Gambar 4.10. Hasil Pencarian “Sistem Informasi”

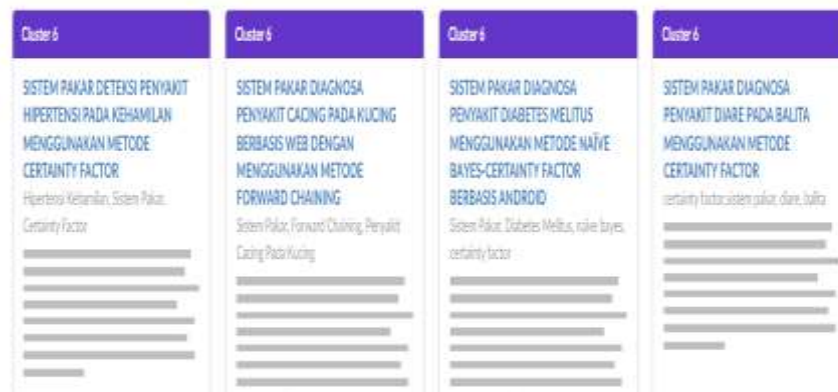
Pada gambar 4.10. topik “Sistem Informasi” digunakan 4 mahasiswa sebagai judul skripsi terletak pada cluster 4. Pada gambar diatas topik “Sistem Informasi” digunakan 4 mahasiswa sebagai judul skripsi dengan masing-masing metode yang berbeda. Objek yang digunakan pada 4 judul diatas juga berbeda-beda antara satu dengan yang lainnya. Topik “Sistem Informasi” adalah topik yang paling banyak digunakan mahasiswa Prodi Informatika Unkhair sebagai judul skripsi. Cluster 4 ditandai dengan garis warna orens.



Gambar 4.11. Hasil Pencarian “Implementasi”

Pada gambar 4.11. topik “Implementasi” digunakan 2 mahasiswa sebagai judul skripsi yang terletak pada cluster 5. Pada gambar diatas topik “Implementasi” digunakan 2 mahasiswa sebagai judul skripsi dengan metode yang sama yaitu K-Nearest Neighbor dan

juga melakukan klasifikasi. Objek yang digunakan pada 2 judul diatas berbeda-beda antara satu dengan yang lainnya. Cluster 5 ditandai dengan garis merah.



Gambar 4.12. Hasil Pencarian “Sistem Pakar”

Gambar 4.12. Hasil Pencarian “Sistem Pakar” Pada gambar 4.7.7. topik “Sistem Pakar” digunakan 4 mahasiswa sebagai judul skripsi yang terletak pada cluster 1. Pada gambar diatas topik “Sistem Pakar” digunakan 4 mahasiswa sebagai judul skripsi dengan metode yang sama yaitu certainty factor dan forward chaining. Objek yang digunakan pada 4 judul diatas berbeda-beda yaitu deteksi berbagai macam jenis penyakit. Cluster 7 ditandai dengan ungu tua.



Gambar 4.13. Hasil Pencarian “Sistem Pendukung Keputusan”

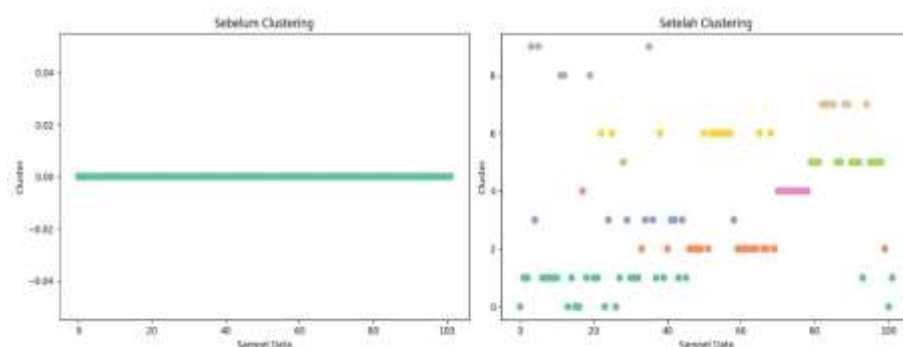
Pada gambar 4.13. topik “Sistem Pendukung Keputusan” digunakan 2 mahasiswa sebagai judul skripsi yang terletak pada cluster 7. Objek yang digunakan pada 2 judul diatas

berbeda-beda yaitu penentuan remisi narapidana dan pembelian mobil baru. Cluster 7 ditandai dengan ungu tua.

4.7. Grafik

4.7.1. Scatter Plot 2 Dimensi

Tahapan visualisasi data pertama yaitu dengan menggunakan *Scatter Plot* 2 Dimensi. *Scatter plot* 2 Dimensi dalam HAC (*Hierarchical Agglomerative Clustering*) adalah cara untuk memvisualisasikan hasil pengelompokan hierarki aglomeratif dari suatu dataset. HAC adalah metode pengelompokan di mana setiap titik data awal dianggap sebagai satu kelompok dan secara bertahap digabungkan menjadi kelompok yang lebih besar hingga satu kelompok utama terbentuk. Grafik *Scatter Plot* 2 dimensi dapat dilihat pada gambar 4.14.



Gambar 4.14. *Scatter Plot* 2 Dimensi

1. Sumbu X dan Y

Sumbu X dan Y pada scatter plot mewakili dua dimensi dari data yang diamati atau dua fitur tertentu yang digunakan dalam proses pengelompokan.

2. Titik Data

Setiap titik pada plot mewakili satu data point (misalnya, suatu pengamatan atau entitas) dalam dataset. Titik-titik tersebut diberi warna atau simbol yang berbeda untuk

menunjukkan kelompok atau klaster ke mana mereka termasuk.

3. Warna atau Simbol Kelompok

Warna atau simbol yang digunakan pada titik-titik tersebut mencerminkan hasil dari proses HAC. Titik-titik yang memiliki warna atau simbol yang sama cenderung termasuk dalam kelompok yang sama.

4. Hierarki Aglomeratif

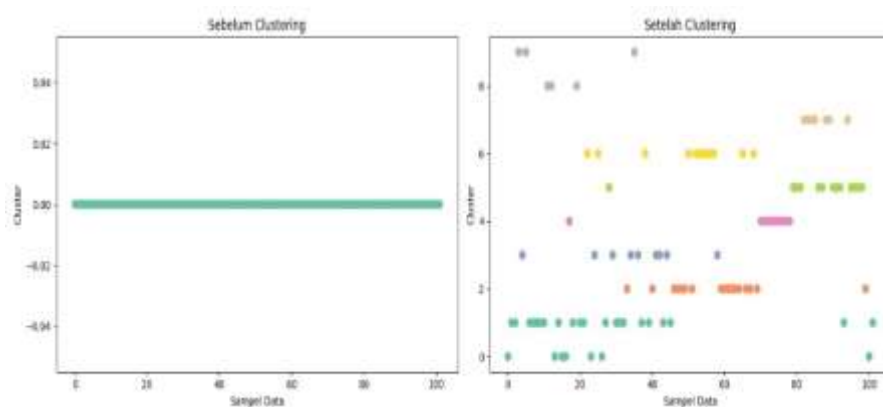
Scatter plot tersebut juga dapat memberikan informasi tentang hierarki aglomeratif. Misalnya, titik-titik yang lebih dekat satu sama lain di plot mungkin lebih dekat dalam hubungan hierarki aglomeratif, menunjukkan bahwa mereka lebih sering digabungkan selama proses pengelompokan.

5. Ukuran Titik

Ukuran titik dapat digunakan untuk menunjukkan beberapa informasi tambahan, seperti beratnya titik tersebut dalam proses pengelompokan atau seberapa signifikan suatu kelompok.

4.7.2. Scatter Plot 3 Dimensi

Grafik *Scatter Plot* 3 dimensi dapat dilihat pada gambar 4.15.



Gambar 4.15. *Scatter Plot* 3 Dimensi

Scatter plot 3D dalam konteks *Hierarchical Agglomerative Clustering* (HAC) adalah cara untuk memvisualisasikan hasil pengelompokan hierarki aglomeratif dari suatu dataset dengan menggunakan tiga dimensi, bertujuan memahami struktur hierarki pengelompokan dalam tiga dimensi, melibatkan sumbu X, Y, dan Z.

1. Sumbu X, Y, dan Z

Sumbu X, Y, dan Z pada scatter plot 3D mewakili tiga dimensi dari data atau tiga fitur tertentu yang digunakan dalam proses pengelompokan.

2. Titik Data

Setiap titik pada plot mewakili satu data point dalam dataset. Titik-titik ini diberi warna atau simbol yang berbeda untuk menunjukkan kelompok atau klaster ke mana mereka termasuk.

3. Warna atau Simbol Kelompok

Warna atau simbol yang digunakan pada titik-titik mencerminkan hasil dari proses HAC. Titik-titik dengan warna atau simbol yang sama mungkin termasuk dalam kelompok yang sama.

4. Hierarki Aglomeratif

Scatter plot 3D juga dapat memberikan informasi tentang hierarki aglomeratif. Jarak relatif antar titik dalam plot mencerminkan hubungan hierarki antar kelompok, di mana titik-titik yang lebih dekat satu sama lain mungkin lebih dekat dalam hubungan hierarki aglomeratif.

5. Rotasi Plot

Karena kita berurusan dengan tiga dimensi, rotasi plot dapat memberikan perspektif yang berbeda terhadap hubungan antar kelompok dan struktur hierarkinya.

6. Ukuran Titik atau Garis

Ukuran titik atau ketebalan garis dapat digunakan untuk menyampaikan beberapa informasi tambahan, seperti seberapa signifikan suatu kelompok atau hubungan antar kelompok.

4.8. Dendrogram

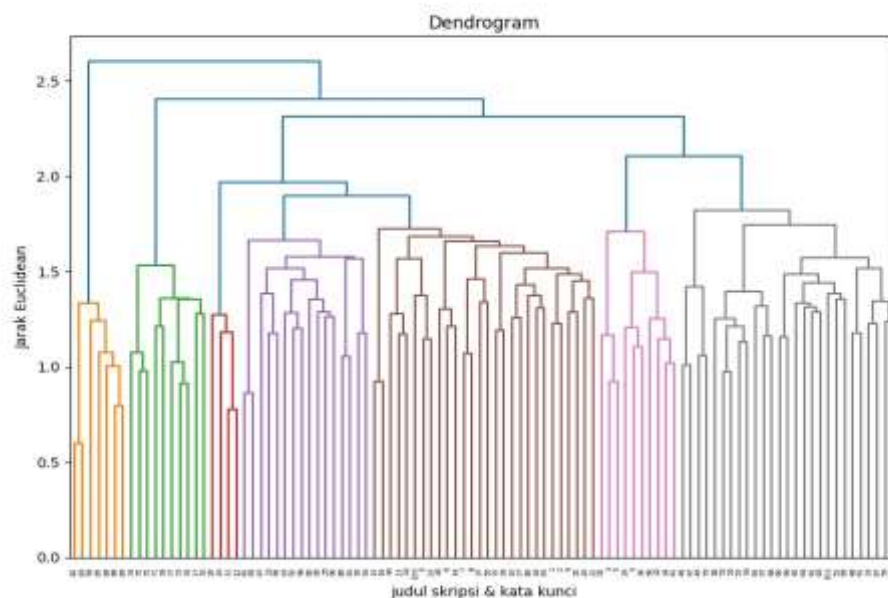
Pada tahap ini dilakukan visualisasi data kedua dengan menampilkan dendrogram. Kunci untuk menafsirkan dendrogram adalah dengan fokus pada ketinggian di mana dua objek digabungkan. Dendrogram membantu dalam memahami urutan penggabungan atau pembentukan kelompok, dimulai dari titik awal hingga pembentukan kelompok yang lebih besar.

Dendrogram di bawah adalah representasi visual dari hierarki *clustering*. Setiap cabang pada dendrogram mewakili pembentukan cluster yang semakin besar. Dengan memeriksa dendrogram, dapat diidentifikasi hubungan hierarki antara cluster dan sub-cluster.

Pada dendrogram di bawah, ketinggian dendrogram menunjukkan urutan penggabungan cluster. Garis berwarna pada dendrogram di bawah berfungsi sebagai petunjuk pengelompokan atau penggabungan kluster yang berbeda. Ketika dua kluster atau lebih digabungkan, ini tercermin dalam bentuk garis pada dendrogram. Warna yang berbeda biasanya digunakan untuk menyoroti kluster yang berbeda yang digabungkan pada tingkat tertentu. Saat dua kluster bergabung, garis yang menghubungkannya akan berbeda warna untuk menunjukkan pengelompokan pada level tertentu. Cara ini digunakan untuk membantu visualisasi dan memahami proses pengelompokan pada dendrogram, serta dipakai untuk menentukan jumlah cluster yang diinginkan yang nanti tetap harus di uji oleh

Silhouette Score.

Dendrogram yang lebih informatif dapat dibuat dengan ketinggian yang mencerminkan jarak antar cluster seperti yang ditunjukkan di bawah ini. Visualisasi dendrogram juga bertujuan untuk membantu menentukan jumlah cluster yang tepat. Cluster yang dihasilkan dari proses clustering menggunakan algoritma *Hierarchical Agglomerative Clustering* sebanyak 2 cluster dengan total 103 judul skripsi dan kata kunci. Yang dapat dilihat pada gambar 4.16.



Gambar 4.16. Visualisasi data dalam bentuk dendrogram

4.9. Word Cloud

Metode visualisasi terakhir yang digunakan untuk memvisualisasikan data adalah dengan menggunakan *wordcloud*. *Word cloud* diatas adalah visualisasi yang menampilkan kata-kata dari teks yang diberikan, dengan ukuran font yang lebih besar untuk kata-kata yang lebih sering muncul dalam teks tersebut. *Word cloud* digunakan untuk menunjukkan kata-kata yang paling penting atau sering muncul dalam teks yang dianalisis dan dapat

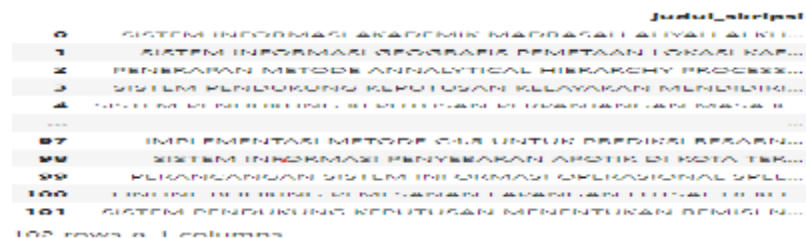
dapat dihitung jarak antar judul dan antar kata kunci. Misalnya, menggunakan metode penghitungan jarak *euclidean distance*.

2. Pembentukan Matriks Kemiripan

Membuat matriks yang memuat jarak antar judul-judul skripsi berdasarkan kesamaan kata atau makna kata.

3. Penerapan Metode HAC

Menggunakan algoritma HAC untuk mengelompokkan judul-judul skripsi menjadi *cluster*. Langkah ini melibatkan penggabungan judul-judul yang paling mirip secara bertahap, lalu implementasi kesamaan linguistik diterapkan di dalam program berupa setiap *cluster* menampilkan beberapa kata yang mirip atau sama dan dikelompokkan sehingga terjadi proses *clustering*, hasilnya dapat dilihat pada gambar 4.18.



	Judul_skripsi
0	SISTEM INFORMASI AKADEMIK MADRASAH ALYATI AL K...
1	SISTEM INFORMASI GEOGRAFI PEMETAAN LOKASI KAP...
2	PENERAPAN METODE ANALYTICAL HIERARCHY PROCES...
3	SISTEM PENDUKUNG KEPUTUSAN KELAYAKAN MENUNDA...
4	ANALISIS KEPUTUSAN MENUNDA KEPUTUSAN MENUNDA...
...	...
97	IMPLEMENTASI METODE C4.5 UNTUK PREDIKSI BERSAR...
98	SISTEM INFORMASI PENYERAPAN AIRSIS DI KOTA TER...
99	PERANCANGAN SISTEM INFORMASI OPERASIONAL SPES...
100	IMPLEMENTASI METODE C4.5 UNTUK PREDIKSI BERSAR...
101	SISTEM PENDUKUNG KEPUTUSAN MENUNDA KEPUTUSAN...

105 rows x 1 columns

Gambar 4.18. Implementasi Kesamaan Linguistik

4. Visualisasi Grafik, Dendrogram dan WordCloud

Menghasilkan Grafik Dendrogram dan *Word Cloud* untuk memvisualisasikan hierarki klaster yang terbentuk dari judul skripsi.

5. Penentuan Jumlah *Cluster*

Dari dendrogram, menentukan jumlah *cluster* yang memadai berdasarkan interpretasi linguistik dari struktur *cluster* dan tingkat kesamaan kata

4.11. Pengujian *Sillhoute Score*

Silhouette Score berkisar antara -1 hingga 1, dan semakin mendekati 1, semakin baik kualitas *clustering*nya. Sebaliknya, nilai yang mendekati 0 atau negatif menandakan bahwa *clustering* tidak optimal dan mungkin tidak memiliki struktur yang jelas. Nilai mendekati 1 menunjukkan bahwa objek dalam suatu klaster memiliki jarak yang jauh lebih dekat dengan objek-objek dalam klaster yang sama daripada klaster lain, menunjukkan pembagian yang baik. Nilai mendekati 0 menunjukkan bahwa ada tumpang tindih antara klaster. Nilai mendekati -1 menunjukkan bahwa objek-objek dalam suatu klaster memiliki jarak yang lebih dekat dengan klaster lain daripada klaster yang sama.

Nilai *Sillhoute Score* dari hasil program diatas dengan menggunakan metode *Hierarc hical Agglomerative Clustering* dengan 8 cluster adalah 0.0733. Nilai 8 dipilih sebagai nilai yang paling ideal karna memiliki nilai *sillhoute* yang paling besar.

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan dengan menerapkan metode *Hierarchical Agglomerative Clustering* dalam clustering judul skripsi mahasiswa pada Prodi Informatika, dapat disimpulkan sebagai berikut:

1. Meskipun metode *Hierarchical Agglomerative Clustering* dapat diimplementasikan dalam *clustering* dokumen skripsi dengan menggunakan perhitungan jarak *Euclidean distance*, hasil yang diperoleh tidak memadai, dengan nilai *sillhouette* hanya mencapai 0.04852. Ini menunjukkan bahwa metode ini tidak memberikan pemisahan kluster yang baik atau interpretasi yang kuat terhadap data yang dianalisis.
2. Penelitian ini memberikan indikasi bahwa pendekatan yang digunakan tidak berhasil menemukan pola atau tren yang jelas dalam judul skripsi. Kontribusi teoritis yang diharapkan dalam membantu memperkaya pemahaman tentang klasifikasi topik dalam literatur akademis menjadi terbatas.
3. Hasil penelitian ini menunjukkan bahwa pengelompokan judul skripsi ke dalam tema atau topik penelitian tertentu tidak dapat dilakukan dengan efektif. Dengan hasil yang kurang memuaskan. Penelitian lebih lanjut dengan metode atau pendekatan lain diperlukan untuk mencapai hasil yang lebih baik.

5.2. Saran

1. Penambahan Fitur Ekstraksi: Selain menggunakan TF-IDF, fitur lain seperti penambahan kata, abstrak atau kata kunci tambahan dapat digunakan untuk memperkaya representasi dokumen, yang mungkin membantu dalam pengelompokan yang lebih akurat.

2. Penggunaan Data yang Lebih Luas, dengan menambah jumlah data atau mencakup lebih banyak judul skripsi dari berbagai topik, analisis clustering mungkin bisa memberikan hasil yang lebih representatif dan bermanfaat.
3. Kombinasi dengan Teknik *Supervised Learning* untuk mengklasifikasikan judul skripsi ke dalam kategori yang sudah ditentukan untuk hasil yang lebih presisi.

DAFTAR PUSTAKA

- Adhe, D., 2020, Implementasi *Text Mining* Pengelompokan Dokumen Skripsi Menggunakan Metode *K-Means Clustering Implementation Of Text Mining For Grouping Thesis Documents Using K-Means Clustering*. Jurnal Eksponensial, 11(2).
- Aditya Wicaksana, D., 2018, *Clustering* Dokumen Skripsi Dengan Menggunakan *Hierarchical Agglomerative Clustering* Vol. 2, Issue 1.
- Akromunnisa, K., 2019, Klasifikasi Dokumen Tugas Akhir (SKRIPSI) Menggunakan *K-Nearest Neighbor*. In *JISKa* Vol. 4, Issue 1.
- Ariadi, D. 2015, Klasifikasi Berita Indonesia Menggunakan Metode *Naive Bayesian Classification* dan *Support Vector Machine* dengan *Confix Stripping Stemmer*. Jurnal Sains Dan Seni ITS Vol. 4, No.2, 4(2), 248–253.
- Cahyani, L., 2022, *Text Mining* untuk Pengelompokan Skripsi di Prodi Pendidikan Informatika Universitas Trunojoyo Madura. In *Jurnal Ilmiah Edutic* Vol. 8, Issue 2.
- Firmansyah, M., 2020, Klasifikasi Kalimat Ilmiah Menggunakan *Recurrent Neural Network*. *Prosiding The 11th Industrial Research Workshop and National Seminar*, 11(1), 488–495.
- Kambey., 2020, Penerapan *Clustering* pada Aplikasi Pendeteksi Kemiripan Dokumen Teks Bahasa Indonesia. 1–8.
- Khaosaroh, S., 2019, Penerapan metode *hierarchical agglomerative clustering* berbasis *single linkage* untuk pengelempokan judul skripsi. 9, 53–64.
- Setiawan, A., 2020, Klasifikasi Artikel Berita Bahasa Indonesia Dengan *Naive Bayes Classifier*. Jurnal Infra, 8(1), 146–151.
- Unang, Achlison., 2020, Analisis Implementasi Pengukuran Suhu Tubuh Manusia dalam Pandemi Covid-19 di Indonesia. *Pixel :Jurnal Ilmiah Komputer Grafis*, 13(2), 102–106.
- Wahyono, T., 2018, *Fundamental of Python for Machine Learning*: Dasar-Dasar Pemrograman *Python* untuk *Machine Learning* dan Kecerdasan Buatan. Gava Media,
- Widya Utami, N., 2022, *Text Mining Clustering* Untuk Pengelompokan Topik Dokumen Penelitian Menggunakan Algoritma *K-Means* Dengan *Cosine Similarity*. In *JINTEKS* Vol. 4, Issue 3.

LAMPIRAN

SOURCE CODE

```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.cluster import AgglomerativeClustering
import matplotlib.pyplot as plt
import scipy.cluster.hierarchy as sch
import joblib
from collections import defaultdict
from sklearn.metrics import silhouette_score
from wordcloud import WordCloud
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from collections import Counter
from PIL import Image
from sklearn.metrics.pairwise import euclidean_distances, cosine_similarity
def create_wordcloud(dataWord, nama_file):
    cluster_label_word_cloud = clusters
    cluster_dict_word_cloud = {}
    for i, label in enumerate(cluster_label_word_cloud):
        if label not in cluster_dict_word_cloud:
            cluster_dict_word_cloud[label] = []
            cluster_dict_word_cloud[label].append(dataWord[i])
    for cluster_dict_word_cloud, judul_list in cluster_dict.items():
        wordcloud = WordCloud(width=800, height=400, background_color='white', font_path=None).generate('
        '.join(judul_list))
    plt.figure(figsize=(10, 5))
    plt.imshow(wordcloud, interpolation='bilinear')
```

```

plt.title(f'Word Cloud Kluster {cluster_dict_word_cloud}')
plt.axis('off')
plt.savefig(f'wordcloud_{nama_file}_{cluster_dict_word_cloud}.png')
plt.show()

def create_dendrogram(tfidf_matrix, title):
plt.figure(figsize=(10, 7))
dendrogram = sch.dendrogram(sch.linkage(tfidf_matrix.toarray(), method='ward'))
plt.title('Dendrogram')
plt.xlabel(title)
plt.ylabel('Jarak Euclidean')
plt.show()

def create_hac(judul_skripsi, kata_kunci):
for i, (judul,kata_kunci) in enumerate(zip(judul_skripsi,kata_kunci)):
cluster_label = hac_model.labels_[i]
judul_dan_kata_kunci = judul + " ( " + kata_kunci + " ) "
cluster_dict[cluster_label].append(judul_dan_kata_kunci)
for cluster_label, judul_list in cluster_dict.items():
print(f"Kluster {cluster_label}:")
for judul in judul_list:
print(judul)
print()

def create_filetext(nama_file,judul_skripsi, kata_kunci):
from collections import defaultdict
cluster_dict_txt = defaultdict(list)
for i, (judul,kata_kunci) in enumerate(zip(judul_skripsi,kata_kunci)):
cluster_label_txt = hac_model.labels_[i]
judul_dan_kata_kunci = judul + " ( " + kata_kunci + " ) "
cluster_dict_txt[cluster_label_txt].append(judul_dan_kata_kunci)

```

```

with open(nama_file, 'w') as file:
    for cluster_label_txt, judul_list in cluster_dict_txt.items():
        file.write(f"Kluster {cluster_label_txt}:\n")
        for judul in judul_list:
            file.write(judul + "\n")
        file.write("\n")

print(f"Data Kluster telah ditulis ke {nama_file}")

data = pd.read_excel('dataset_judul_skripsi.xlsx')
dataWord = pd.read_excel('dataset_judul_skripsi.xlsx')
tfidf_vectorizer = TfidfVectorizer(stop_words='english')
tfidf_matrix_judul_skripsi = tfidf_vectorizer.fit_transform(data['judul_skripsi'])
tfidf_matrix_kata_kunci = tfidf_vectorizer.fit_transform(data['kata_kunci'])
tfidf_matrix = tfidf_vectorizer.fit_transform(data['judul_skripsi'], data['kata_kunci'])

num_clusters = 8

hac_model = AgglomerativeClustering(n_clusters=num_clusters, metric='euclidean', linkage='ward')
# hac_model = AgglomerativeClustering(n_clusters=num_clusters)

clusters = hac_model.fit_predict(tfidf_matrix.toarray())

cluster_sizes = np.bincount(clusters)

data['cluster'] = clusters

cluster_dict = defaultdict(list)

for i, (judul, kata_kunci) in enumerate(zip(data['judul_skripsi'], data['kata_kunci'])):
    cluster_label = hac_model.labels_[i]

    judul_dan_kata_kunci = judul + " ( " + kata_kunci + " ) "

    cluster_dict[cluster_label].append(judul_dan_kata_kunci)

fjson = pd.DataFrame(data)

# Convert the DataFrame to JSON and save it to a file
dfjson.to_json("data.json", orient="records")

print("DataFrame has been converted to JSON and saved in data.json")

```

```

create_dendogram(tfidf_matrix_judul_skripsi, 'judul skripsi')
create_dendogram(tfidf_matrix_kata_kunci, 'kata kunci')
create_dendogram(tfidf_matrix, 'judul skripsi & kata kunci')

# Langkah: Proses data
data_plot = data[['judul_skripsi']].dropna()

# Langkah: Pembuatan Matriks Jarak sebelum Clustering
vectorizer_before = TfidfVectorizer()
tfidf_matrix_before = vectorizer_before.fit_transform(data_plot['judul_skripsi'])
distance_matrix_before = 1 - cosine_similarity(tfidf_matrix_before)

# Langkah 4: Hierarchical Agglomerative Clustering sebelum Clustering
n_clusters_before = 1 # Ubah sesuai dengan jumlah cluster yang diinginkan sebelum clustering
cluster_before = AgglomerativeClustering(n_clusters=n_clusters_before, linkage='ward', metric='euclidean')
cluster_labels_before = cluster_before.fit_predict(distance_matrix_before)

# Langkah 5: Pembuatan Matriks Jarak setelah Clustering
vectorizer_after = TfidfVectorizer()
tfidf_matrix_after = vectorizer_after.fit_transform(data['judul_skripsi'])
distance_matrix_after = 1 - cosine_similarity(tfidf_matrix_after)

# Langkah 6: Hierarchical Agglomerative Clustering setelah Clustering
n_clusters_after = 10 # Ubah sesuai dengan jumlah cluster yang diinginkan setelah clustering
cluster_after = AgglomerativeClustering(n_clusters=n_clusters_after, linkage='ward', metric='euclidean')
cluster_labels_after = cluster_after.fit_predict(distance_matrix_after)

# Langkah 7: Tampilkan Hasil dalam Scatter Plot sebelum dan setelah Clustering
fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(15, 5))

# Scatter Plot Sebelum Clustering
axes[0].scatter(range(len(data)), cluster_labels_before, c=cluster_labels_before, cmap='Set2', s=50)
axes[0].set_xlabel('Sampel Data')
axes[0].set_ylabel('Cluster')
axes[0].set_title('Sebelum Clustering')

```

```

# Scatter Plot Setelah Clustering
axes[1].scatter(range(len(data)), cluster_labels_after, c=cluster_labels_after, cmap='Set2', s=50)
axes[1].set_xlabel('Sampel Data')
axes[1].set_ylabel('Cluster')
axes[1].set_title('Setelah Clustering')
plt.tight_layout()
plt.show()

# Langkah 2: Proses data
data = data[['judul_skripsi']].dropna()

# Langkah 3: Pembuatan Matriks Jarak
vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(data['judul_skripsi'])
distance_matrix = 1 - cosine_similarity(tfidf_matrix)

# Langkah 4: Hierarchical Agglomerative Clustering
n_clusters = 10 # Ubah sesuai dengan jumlah cluster yang diinginkan
cluster = AgglomerativeClustering(n_clusters=n_clusters, metric='euclidean', linkage='ward')
cluster_labels = cluster.fit_predict(distance_matrix)

# Langkah 5: Tampilkan Hasil dalam Scatter Plot 3D
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

# Buat label dari 0 hingga jumlah sampel data
labels = range(len(data))

# Gunakan scatter plot 3D dengan warna berdasarkan label cluster
scatter = ax.scatter(labels, cluster_labels, labels, c=cluster_labels, cmap='Set2', s=50)

# Sesuaikan label dan judul plot
ax.set_xlabel('Data Judul Skripsi')
ax.set_ylabel('Cluster')
ax.set_zlabel('Data Judul Skripsi')

```

```

ax.set_title('3D Scatter Plot hasil Hierarchical Agglomerative Clustering pada Judul Skripsi')
# Tampilkan colorbar sebagai legenda
cbar = plt.colorbar(scatter)
cbar.set_label('Cluster', rotation=270, labelpad=15)
# Tampilkan plot
plt.show()
create_wordcloud(dataWord['judul_skripsi'], 'judul_skripsi')
create_wordcloud(dataWord['kata_kunci'], 'kata_kunci')
import nltk
nltk.download('stopwords')
nltk.download('punkt_tab')
# Menggunakan stopwords untuk bahasa Indonesia dan Inggris
stop_words_indonesia = set(stopwords.words('indonesian'))
# stop_words_english = text.ENGLISH_STOP_WORDS
# Menggabungkan stopwords dari kedua bahasa
stop_words_both = stop_words_indonesia.union(stop_words_english)
# stop_words = set(stopwords.words('english')) # Menggunakan stopwords bahasa Inggris sebagai contoh
# Menambahkan label cluster ke data
dataWord['Cluster'] = clusters
# Tambahkan kolom 'Tokens' ke DataFrame
dataWord['Tokens'] = dataWord['judul_skripsi'].apply(lambda x: [word.lower() for word in word_tokenize(x) if
word.isalnum() and word.lower() not in stop_words_both])
# Mencetak baris yang menyebabkan kesalahan
try:
# Menghitung kata-kata yang paling sering muncul pada setiap cluster
cluster_words = dataWord.groupby('Cluster')['Tokens'].apply(lambda x: Counter(word for sublist in x for word
in sublist).most_common(6))
# Menampilkan hasil
for cluster, words in cluster_words.items():

```

```

print(f'Cluster {cluster + 1}: {words}')
except Exception as e:
    print("Error:", e)
    print("Error Row(s):")
    print(dataWord[dataWord["Tokens"].apply(lambda x: not isinstance(x, list))])
def preprocess_text(texts):
    from sklearn.feature_extraction.text import TfidfVectorizer
    vectorizer = TfidfVectorizer(stop_words='english', max_features=1000)
    X = vectorizer.fit_transform(texts)
    # Normalisasi
    from sklearn.preprocessing import StandardScaler
    scaler = StandardScaler(with_mean=False) # with_mean=False karena TF-IDF sparse
    X_scaled = scaler.fit_transform(X)
    return X_scaled
data = pd.read_excel('dataset_judul_skripsi.xlsx')
X = preprocess_text(data['judul_skripsi'] + data['kata_kunci'])
def hierarchical_clustering(X, n_clusters):
    from sklearn.cluster import AgglomerativeClustering
    model = AgglomerativeClustering(n_clusters=n_clusters, metric='euclidean', linkage='ward')
    labels = model.fit_predict(X.toarray())
    return labels
def find_optimal_clusters(X):
    from sklearn.metrics import silhouette_score
    best_score = -1
    best_n_clusters = 2
    for n_clusters in range(2, 100): # Coba jumlah cluster dari 2 hingga 10
        labels = hierarchical_clustering(X, n_clusters)
        score = silhouette_score(X, labels)

```

```
if score > best_score:
    best_score = score
    best_n_clusters = n_clusters
    print(f'Jumlah Cluster: {n_clusters}, Silhouette Score: {score}')
    return best_n_clusters, best_score

# Cari jumlah cluster optimal
optimal_n_clusters, optimal_score = find_optimal_clusters(X)
print(f'Cluster Paling Optimal: {optimal_n_clusters}, Silhouette Score: {optimal_score}')
```



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024

Pukul : 13:30 - 15:00

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA HIL. ARSAD

NPM : 07351811042

Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- ①. Perbaiki Serapat
- ②. Buat Analisis hasil Clustering & akhir bab 4
- ③. Tambahkan & saran untuk perbaikan preprocessing
dari dan p'gunaan algoritma yg berbeda.

Dosen Pembimbing I,

Ir. ABDUL MUBARAK, S.Kom., M.T., IPM
NIP. 198212062014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024

Pukul : 13:30 - 15:00

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA HI. ARSAD

NPM : 07351811042

Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- Pahami hasil penelitian anda

26 & 2024
Alc

Dosen Pembimbing II,

MUHAMMAD FHADLI, S.Kom., M.Sc.
NIP. 199611232023211012



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024

Pukul : 13:30 - 15:00

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA HI. ARSAD

NPM : 07351811042

Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

① Bab 4, belum ada hasil Clustering.

② Lakukan pengisian ulang nilai silhouette terbank
/ menentukan jumlah Cluster

③ Ditambahkan, nilai dan kutipan pertama

④ Kesimpulan diperbaiki. Perbaiki Kesimpulan
menjawab Rumusan masalah yg ada
pd Bab I -

Dosen Penguji I,


I. AMAL KHAIRAN, S.T., M.Eng., IPM
NIP. 197401112003121003



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024

Pukul : 13:30 - 15:00

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA HI. ARSAD

NPM : 07351811042

Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

1. Perbaiki Aplikasi

2. Daftar Pustaka minimal 15 referensi

Ae 19/7/2024

Dosen Penguji II,

HAIRIL KURNIADI SIRAJUDDIN, S.Kom., M.Kom.
NIP. 198204272023211009



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 01 MARET 2024

Pukul : 13:30 - 15:00

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA HI. ARSAD

NPM : 07351811042

Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Kuasai pemrograman yang digunakan dalam penelitian ini

Analisis algoritma agar mendapatkan kesimpulan kenapa akurasi rendah

Kuasai algoritma yang digunakan (perluhatkan simulasinya)

Kuasai office

14/8/2024

Ara Revini

Dosen Penguji III,

Ir. SAKIN LUTFI S.Kom., M.T., IPM
NIP. 198601112014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 12 JANUARI 2024
Pukul : 13:30 - 15:30
Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA HI. ARSAD
NPM : 07351811042
Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- 1) Tinjau ulang Hasil Analisis
- 2) perbaiki simpulan

23/24
Abdul Mubarak

Dosen Pembimbing I,

Ir. ABDUL MUBARAK, S.Kom., M.T., IPM
NIP. 198212062014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 12 JANUARI 2024

Pukul : 13:30 - 15:30

Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA IHL ARSAD

NPM : 07351811042

Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI

BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- Pahami source code

Handwritten signature and date 23/1/2024

Dosen Pembimbing II.

Handwritten signature of Muhammad Fhadli

MUHAMMAD FHADLI, S.Kom., M.Sc.
NIP. 199611232023211012



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 12 JANUARI 2024

Pukul : 13:30 - 15:30

Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA HI. ARSAD

NPM : 07351811042

Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

- Abstrak Glma (knsi tngs, Paratka, Metode, Hasilnya) ✓
- Tngs paratka dan Rumus masalah tidak sinkron
- Sistematika Penulisan tidak lengkap ✓
- Jurnal paratka itu perlu → harus ✓
- ✓ - Gambar 4.7.3 berikan penjelasan lebih lanjut ✓
- ✓ - Hasil Clusker 1 & Gab 4 tidak ada ✓
- Kesimpulan Gab 5 tidak / belum menjawab Rumus masalah.

Dosen Penguji I,


Ir. AMAL KHAIRAN, S.T., M.Eng., IPM
NIP. 197401112003121003



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 12 JANUARI 2024

Pukul : 13:30 - 15:30

Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : WIDYA MAULINDA HI. ARSAD

NPM : 07351811042

Judul : IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

1. Perbaiki aplikasi
2. Coba simulasikan dalam bentuk nyata dalam bentuk sebuah aplikasi untuk membuktikan output dari hasil ujicoba

Ace 2/2/2024

Dosen Penguji II,

HAJRIL KURNIADI SIRAJUDDIN, S.Kom., M.Kom.
NIP. 198204272023211009



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

: JUMAT, 12 JANUARI 2024

: 13:30 - 15:30

: RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

: WIDYA MAULINDA HI. ARSAD

: 07351811042

: IMPLEMENTASI METODE HIERARCHICAL AGGLOMERATIVE
CLUSTERING UNTUK CLUSTERING DOKUMEN SKRIPSI
BERDASARKAN KESAMAAN LINGUISTIK (STUDI KASUS: PRODI
INFORMATIKA UNKHAIR)

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

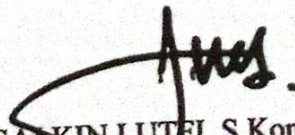
Pelajari algoritma serta buat simulasi di excel metode yang digunakan

Kuasai pemrograman yang menimplementasikan algoritma yang digunakan

22/1/2024

Ale Roveni

Dosen Penguji III,


Ir. SALKIN LUTFI, S.Kom., M.T., IPM
NIP. 198601112014041002



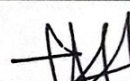
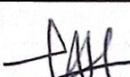
LEMBAR ASISTENSI HASIL

Nama Mahasiswa : Widya Maulinda Hi Arsad
NIM : 07351811042
Dosen Pembimbing I : Ir. Abdul Mubarak, S.Kom., M.T.
Judul : Implementasi Metode Hierarchical Agglomerative Clustering untuk Clustering Dokumen Skripsi berdasarkan Kesamaan Linguistik (Studi Kasus : Prodi Informatika Unkhair)

NO	Tanggal	Uraian	Paraf
1.	20/10/2023	- Jelaskan detail per kluster dari dendrogram - Jelaskan alur dari Silhouette	
2.	01/11/2023	- Tambah Metode Visualisasi lain selain dendrogram - Analisis tentang kesamaan linguistik tambahkan Bab 4 - Arahkan judul sesuai atau belum.	
3.	13/11/2023	- Tambahkan analisis linguistik lebih detail - Detailkan data hasil program	
4.	22/11/2023	- Tambahkan gambar implementasi tf-idf & HAC - Tambah grafik untuk memperjelas cluster - perbaiki linguistik	
5.	29/11/2023	- ganti grafik	
6.	12/12/2023	Acc Hasil Skripsi	



Nama Mahasiswa : Widya Maulinda Hi Arsad
NIM : 07351811042
Dosen Pembimbing II : Muhammad Fhadli, S.Kom., M.Cs.
Judul : Implementasi Metode *Hierarchical Agglomerative Clustering* untuk
Clustering Dokumen Skripsi berdasarkan Kesamaan Linguistik
(Studi Kasus : Prodi Informatika Unkhair)

NO	Tanggal	Uraian	Paraf
1.	05/08/2023	- Tambahkan kata kunci & program - Tambahkan Data (Judul (Kripsi)	
2	19/09/2023	Tambah tabel silhouette	
		Selesaikan bab 4 + 5	
		Selesaikan revisi peguji	
3.	02/10/2023	Perbaiki format	
		Acc	